

# **AMBOSS ORG-M**

**Kurzbeschreibung  
6BB9300-0XX00-0C/04**

**Programmbibliothek  
Karlsruhe**

Titelblatt, Übersicht  
Vorwort, Inhaltsverzeichnis  
Literaturverzeichnis

---

Einführung

---

Verwaltungsobjekte

---

Nahtstellen des ORG

---

Programmorganisation

---

Ein-/Ausgabe-Organisation

---

Dateiorganisation

---

Simulation

---

Zeitbearbeitung

---

Bedien- und Dialogfunktionen

---

Generierfunktionen

---

Systemaktivierungsfunktionen

---

Ladefunktionen

---

Fehlerbehandlung

---

Stichwortverzeichnis

---

Seitenübersicht

5

5

5

5

## Programmbibliothek Karlsruhe

Kurzbeschreibung  
ORG-M

6BB9300 - OXX00 - OC/04

Organisationsprogramm

Stand: August 1989

**Deskriptoren** Betriebssystem, Rechnersysteme SICOMP M, Bürocomputer 6.680, Organisationsprogramm, Programmverwaltung, Speicherorganisation, Ein-Ausgabe-Verwaltung, Dateiverwaltung, Zeitbearbeitung, Bedienung, Wiederanlauf, Laden, Fehlerbearbeitung

**Zusammenfassung** Das ORG-M umfaßt:

- Zentralteil:  
Unterbrechungsanalyse, Wiederanlauf, Neustart, Wiederaufsetzen, Zeitbearbeitung, Fehlerbehandlung, Systemsimulationen für Befehle des optionellen Gleitpunkt-Prozessors, Konvertierungen und Hilfsroutinen. ORG-Segmentierung, Feldübertragung, dynamische Datenfeldverwaltung. Nahtstellenfunktionen.

- Programmorganisation:  
Start, Ende, Zustands- und Platzwechsel, Warten, Anhalten, Fortsetzen. Koordinierungszähler, Programmparameterübergabe. PRP-Organisation, Laufbereichsverwaltung, feste und alternative Laufbereiche, Platzwechselanforderung und -sperre, Zeitscheibenbearbeitung. Common Data, Common Code.

- Ein-Ausgabe-Organisation:  
Standard-Ein-Ausgabe-Aufrufe für Drucker, Sicht- und Erfassungsstation, PSD-Laufwerk. Aufrufe für Peripherenspeichertransfer, Magnetband- und -kassettenbearbeitung, Datenübertragungsperipherie, Digital- und Analog-Ein- und Ausgabe, Alarmbearbeitung, alarmgesteuerte Koordinierung, Stütztreiber, Anwendertreiber, Datenpufferungen. Bedienung und Quittung.

- Dateiorganisation:  
Zugriffsarten seriell und direkt, Lesen und Schreiben von Dateietiketten, Längenänderung, Dateischutz. Datenträger: Platten, Disketten.

- ORG-nahe Programme:  
Standardbedienprogramm einschl. Anlauffunktionen und Systemauskunft. Monitor. Grundsprache-Lader zum Laden und Löschen von Ablaufobjekten und Common Data. Laden segmentierter Programme, einseitiges Binden beim Laden. Arbeitsform-Lader. Grundbetriebssystem, Generierprogramme. Wartungshilfen.

### Technische Daten

Sprache:

Assemblersprache ASS-M und C

Mindestanlagenausstattung,  
Hardware:

Basissysteme SICOMP M bzw. Bürocomputer 6.680 (ZE 01, ZE 02, ZE 03, ZE 04 mit Platten- und Magnetbandkassettenlaufwerken, Datensichtstation). Für ein HSP-ORG genügt eine ZE.

Speicherplatzbedarf:

Abhängig vom Ausbau (Generierung):

HSP:

bis 123 Seiten (bei ZE 02, ZE 03 und ZE 04 auch mehr), Standard-Laufbereich mind. 20 Seiten

PSD:

ab ca. 180 Seiten ohne Wartebereich

1

2

3

4

**Übersicht System-Software** AMBOSS ist die Systemsoftware für die Minicomputer der Rechner-systeme SICOMP M und für die Bürocomputer 6.680 des Systems 6.000. AMBOSS stellt zwei Betriebssysteme zur Verfügung: B S - M und A M B O S S 4 (s. Katalog PR 30, /4/).

Anmerkung: SICOMP ersetzt künftig die bisherige Bezeichnung SIEMENS SYSTEME 300.

**Betriebs-system** Betriebssysteme umfassen die Grundprogramme, die Sprach-systeme und Dienstprogramme/-systeme. Die Grundprogramme: ORG-M inkl. ORG-nahe Programme bilden den gemeinsamen Kern der Betriebssysteme BS-M und AMBOSS 4.

**Organisations-programm** Zum Organisationsprogramm O R G - M gehören die ORG-Bau-steine, der ORG-Generator, die beiden Lader, das Standard-bediensprogramm, das Dienstprogramm MODGEN, die Wartungshilfe EVITAM sowie Schnittstellen für Prozeßverkehr, Datenübertra-gung und Geräte-Sonderverkehr.

Die ORG-nahen Programme / Subsysteme: CAGE-M, LINK-M, MONIM, MBDO-M, AMGENM, SPOOL-M und EVITAM sind nicht innerhalb dieser ORG-Dokumentation beschrieben (s. /4/).

**Übersicht ORG-Dokumentation** Das Dokumentations-Konzept des Organisationsprogramms ORG-M orientiert sich an den Aufgaben des Anwenders während der Projektabwicklung eines Anwendersystems. Die Benutzerober-fläche des ORG-M ist deshalb in Einzelunterlagen beschrieben, die diesen Aufgaben entsprechen (siehe Tabelle unten).

| Unterlagensatz<br>des ORG-M  |                  | geeignet<br>zum ....    | Bewerten       |                  |                  |          |   |   |   |
|------------------------------|------------------|-------------------------|----------------|------------------|------------------|----------|---|---|---|
|                              |                  |                         | Projektieren   |                  |                  | Bedienen |   |   |   |
| Produkt-<br>be-<br>zeichnung | Bestellnummer    | Unterlagenart           | Implementieren | Installieren     | Service          |          |   |   |   |
|                              |                  |                         | ORG-M          | 6BB9300-OXX00-OC | Kurzbeschreibung |          |   |   |   |
| ORG-M                        | 6BB9300-OXX00-OG | Programmierhandbuch     |                |                  |                  |          |   | x |   |
| AMGENM                       | 6BE5430-OXX00-OB | Beschreibung            |                |                  |                  |          | x |   |   |
| ORG-M                        | 6BB9300-OXX00-OK | Inbetriebnahmeanleitung |                |                  |                  |          | x |   |   |
| ORG-M                        | 6BB9300-OXX00-OA | Bedienungshandbuch      |                |                  |                  | x        |   |   |   |
| ORG SOPEM                    | 6BB2018-OXX00-OB | Beschreibung            |                |                  |                  |          |   | x | x |
| ORG SOAM                     | 6BB2013-OXX00-OG | Programmierhandbuch     |                |                  |                  |          |   | x |   |
| ORG-M                        | 6BB9300-OXX00-OL | Wartungshandbuch        |                |                  |                  | x        |   |   |   |

Zielsetzung, Aufbau, Verbindlichkeit, Formales

- Zielsetzung** Diese Beschreibung enthält in knapper Form die Leistungen des Organisationsprogramms ORG-M. Sie ist auf Anwenderbelange ausgerichtet (Anwendersicht) und stellt die Funktionen und Eigenschaften im Gesamtzusammenhang dar.
- Die Kurzbeschreibung vermittelt bei Akquisition und Einsatzplanung einen Überblick zum ORG-Leistungsumfang und erleichtert die Bewertung des ORG-M.
- Aufbau** Die Beschreibung enthält im Abschnitt 1 eine System-einführung und die Leistungen/Eigenschaften in Aufzähl-form. Letzteres zum raschen Informieren für den Kenner. Die Abschnitte 2 bis 13 beschreiben das ORG und seine Leistungen thematisch gegliedert.
- Den Zugriff auf Einzelaussagen erleichtern Inhaltsver- zeichnis und Literaturverzeichnis am Anfang sowie das Stichwortverzeichnis am Ende der Unterlage.
- Abgrenzung** Die Kurzbeschreibung gibt die Leistungen der aktuellen Ausbaustufe an und nimmt keinen Bezug auf bisherige Systeme.
- Zu Hardware-Leistungen/Eigenschaften der Modellreihen sei auf die einschlägigen Unterlagen verwiesen.
- Verbindlichkeit, Produktinformation** Die Programmbibliothek liefert mit dem Datenträger des Organisationsprogramms eine Produktinformation. Letztere enthält eventuelle Fehler, Sonderfälle, Ergän- zungen, Einschränkungen oder Abänderungen der Beschrei- bung sowie weitere Betriebshinweise. Sie ist als separa- ter Bestandteil dieser Beschreibung aufzufassen und ihr in Zweifelsfällen in der Verbindlichkeit übergeordnet.
- Modellreihen** Rechnersystem SICOMP M bietet zur Zeit die Modelle: SICOMP M20, SICOMP M25, SICOMP M26, SICOMP M30, SICOMP M50, SICOMP M56, SICOMP M60, SICOMP M70, SICOMP M76 und SICOMP M80 (s. /4/); System 6.000 die entsprechenden Bürocomputer 6.68x-yyyy. Die Aussagen zu den Unterlagen zu ORG-M gelten für alle Modelle, soweit die Hardwarekonfiguration dies ermöglicht. Ab- weichungen hiervon sind angegeben.
- Formales** Eine Seitenübersicht steht am Ende. Verweise auf andere Abschnitte erfolgen durch Angabe der Abschnittsnummer (Beispiel: s. 4.1). Literatur ist so /nummer/ zitiert.
- Änderungen** Beschreibungs-Änderungen können Neuauflagen, Austausch- blätter (inkl. Schachtelseiten, z.B. Seite 79.1) oder Hinweise in der Produktinformation zur Folge haben.
- Benutzervorschläge** Aus dem Anwenderkreis sind Vorschläge, Hinweise oder Korrekturen zu dieser Beschreibung auf dem vorbereite- ten Formular erwünscht (siehe letztes Blatt).

---

| <u>Inhaltsverzeichnis</u>                                | <u>Seite</u> |
|--|--------------|
| Übersicht, Systemsoftware, ORG-Dokumentation             | 0.1          |
| Vorwort  | 0.2          |
| Inhaltsverzeichnis                                       | 0.3          |
| Literaturverzeichnis                                     | 0.6          |
| <br>   |              |
| 1 <u>Einführung</u>                                      | 1            |
| 1.1      Interne Funktionen, Hauptspeicher, Adressierung | 4            |
| 1.2      Leistungsmerkmale                               | 5            |
| 1.2.1    Struktur- und Betriebsparameter                 | 5            |
| 1.2.2    Ablauf- und Verwaltungsobjekte                  | 6            |
| 1.2.3    Funktionen                                      | 9            |
| 1.2.4    Unterbrechungssystem                            | 12           |
| 1.2.5    Ein-Ausgabe-Funktionen                          | 13           |
| 1.2.6    Sicherungsfunktionen                            | 15           |
| 1.2.7    Hilfs- und Zusatzfunktionen                     | 17           |
| 1.2.8    Aufrufe und ORG-Kommandos                       | 18           |
| 1.3      Identifikationskennungen                        | 22           |
| <br>   |              |
| 2 <u>Verwaltungsobjekte</u>                              | 23           |
| 2.1      Ablaufobjekte, Programme                        | 23           |
| 2.2      Geräte  | 25           |
| 2.3      Dateien, Bibliotheken                           | 27           |
| 2.4      Gemeinsame Daten und Felder                     | 28           |
| 2.5      Pakete  | 29           |
| 2.6      Laufbereiche, Wartebereich                      | 30           |
| <br>   |              |
| 3 <u>Nahtstellen des ORG</u>                             | 31           |
| 3.1      Aufrufe   | 31           |
| 3.2      Testnahtstelle                                  | 33           |
| 3.3      Monitornahtstelle                               | 33           |
| 3.4      Anwendernahtstellen                             | 34           |

---

|       |  |      |
|-------|--|------|
| 4     | <u>Programmorganisation</u>                            | 35   |
| 4.1   | Buchführung  | 35   |
| 4.2   | Programmzustände und Zustandswechsel                   | 36   |
| 4.3   | Platzwechsel   | 37   |
| 4.4   | Koordinierungsfunktionen                               | 39   |
| 4.4.1 | Programm starten und beenden                           | 39   |
| 4.4.2 | Programm unterbrechen und wieder fortsetzen            | 40   |
| 4.4.3 | Auf Operationsende warten                              | 41   |
| 4.4.4 | Koordinierungszähler                                   | 42   |
| 4.5   | Programm- und Platzwechselfunktionen                   | 44   |
| 4.6   | Strukturierung und Benutzung der alternat.Laufbereiche | 45   |
| 4.7   | Parameterübergabe, Objektzugriffe                      | 47   |
| 4.8   | Datenaustausch zwischen Programmen                     | 48   |
| 4.8.1 | Datenaustausch über Common Data                        | 49   |
| 4.8.2 | Datenübergabe in Registern                             | 49   |
| 4.8.3 | Datenaustausch durch Feldübertragung                   | 50   |
| 4.9   | Segmentierte Programme                                 | 51   |
| 5     | <u>Ein-/Ausgabe-Organisation</u>                       | 53   |
| 5.1   | Koordinierung des Ein-/Ausgabe-Verkehrs                | 54   |
| 5.2   | Standard-Ein-/Ausgabe                                  | 54   |
| 5.3   | Standardgeräteverkehr                                  | 56   |
| 5.3.1 | Ein/Ausgabeeinheiten                                   | 56.2 |
| 5.3.2 | Periphere Speichereinheiten                            | 56.2 |
| 5.3.3 | Datenübertragungssteuerungen (DUST)                    | 58   |
| 5.4   | Datenpufferung   | 61   |
| 5.5   | Verkehr mit Prozeßperipherie                           | 63   |
| 5.5.1 | Funktionen für "schnelle" Prozeßsignalformer           | 64   |
| 5.5.2 | Funktionen zur Alarmbearbeitung                        | 64   |
| 5.5.3 | Integrierende Analogeingabe                            | 65   |
| 5.6   | Anwendertreiber  | 66   |
| 6     | <u>Dateiorganisation</u>                               | 67   |
| 6.1   | Dateiorganisatorische Funktionen                       | 69   |
| 6.2   | Datei-Ein-/Ausgabe, PSD                                | 71   |
| 6.3   | Datenträgerwechsel im Betrieb, PSD                     | 72   |



---

|       |  |     |
|-------|--|-----|
| 7     | <u>Simulation</u>                      | 73  |
| 8     | <u>Zeitbearbeitung</u>                 | 75  |
| 8.1   | Hardware-Einrichtungen                 | 75  |
| 8.2   | Software-Einrichtungen                 | 76  |
| 8.2.1 | Zeitmodifizierte Aufrufe               | 77  |
| 8.2.2 | Relativzeitzähler                      | 78  |
| 9     | <u>Bedien- und Dialogfunktionen</u>    | 79  |
| 9.1   | Bedienungsablauf                       | 80  |
| 9.2   | ORG-Bedienungen                        | 82  |
| 9.3   | Standardbedienprogramme                | 82  |
| 9.4   | Bedienbare Anwenderprogramme           | 84  |
| 9.5   | Programmierte Bedienung                | 85  |
| 9.6   | Bedienzeichenersatz                    | 86  |
| 10    | <u>Generierfunktionen</u>              | 87  |
| 11    | <u>Systemaktivierungsfunktionen</u>    | 91  |
| 11.1  | Wiederanlauf                           | 92  |
| 11.2  | Neustart                               | 94  |
| 11.3  | Wiederaufsetzen                        | 95  |
| 11.4  | Spannungsausfall am Erweiterungsrahmen | 96  |
| 12    | <u>Ladefunktionen</u>                  | 97  |
| 12.1  | Aufbau der Ladeobjekte                 | 99  |
| 12.2  | Ladeparameter                          | 100 |
| 12.3  | Lade/Lösch-Funktionen                  | 100 |
| 12.4  | Platzvergabe beim Laden                | 105 |
| 13    | <u>Fehlerbehandlung</u>                | 107 |
|       | Stichwortverzeichnis                   | 111 |
|       | Seitenübersicht                        | 121 |
|       | Benutzervorschläge                     | -   |

Literaturverzeichnis für Systemsoftware-Anwenderbeschreibungen

|   |  | Bestellnummer                    |
|---|--|----------------------------------|
| -----<br>  Grundlegendes und übergeordnetes Schrifttum, Normen  <br>----- |  |                                  |
| /1/   | KOSS, Kommunikationsschnittstelle, Beschreibung  | E80850-A216-X                    |
| /2/   | KOMPAT ORG PV/M  | 6BB9300-OXX00-OF                 |
| /3/   | Ausbildungsprogramm<br>Schule für Automatisierung,<br>Rechnersysteme und Anwendungen   | beziehbar bei<br>E881, Karlsruhe |
| /4/   | Katalog PR 30, SICOMP M, Minicomputer  | E86010-K6130-A101                |
| /5/   | Befehlstabelle, Unterbrechungsstruktur,<br>EA-Verkehr (ZE01/02/03/04)<br>Schule für Automatisierung, Rechnersysteme<br>und Anwendungen | beziehbar bei<br>E881, Karlsruhe |
| /6/   | Koordinierungszähler-Anwendungsbeispiele ANKOR   | P71100-B3055-X-X-35              |
| /8/   | SICOMP M Betriebsanleitungen   |                                  |
|   | SICOMP M20<br>Handbuch Betriebsanleitung   | C79000-H8700-C358                |
|   | SICOMP M25<br>Handbuch Betriebsanleitung   | Anfrage E885<br>Nbg-Moorenbrunn  |
|   | SICOMP M26<br>Handbuch Betriebsanleitung   | Anfrage E885<br>Nbg-Moorenbrunn  |
|   | SICOMP M30<br>Betriebsanleitung Sammelmappe Teil 1   | C79000-G8700-C95                 |
|   | SICOMP M50<br>Betriebsanleitung Sammelmappe Teil 1   | Anfrage E885<br>Nbg-Moorenbrunn  |
|   | SICOMP M56<br>Betriebsanleitung Sammelmappe Teil 1   | Anfrage E885<br>Nbg-Moorenbrunn  |
|   | SICOMP M70<br>Betriebsanleitung Sammelmappe Teil 1   | C79000-G8700-C98                 |
|   | SICOMP M76<br>Betriebsanleitung Sammelmappe Teil 1   | Anfrage E885<br>Nbg-Moorenbrunn  |
|   | SICOMP M80<br>Betriebsanleitung Sammelmappe Teil 1   | Anfrage E885<br>Nbg-Moorenbrunn  |

---

|     |                               |                    |
|-----|-------------------------------|--------------------|
| /9/ | SICOMP M Systembeschreibungen |                    |
|     | SICOMP M20                    | E80850-A200-X-X    |
|     | SICOMP M25                    | Anfrage E885 Nbg-M |
|     | SICOMP M26                    | Anfrage E885 Nbg-M |
|     | SICOMP M30                    | E80850-A75-X-X     |
|     | SICOMP M50                    | Anfrage E885 Nbg-M |
|     | SICOMP M56                    | Anfrage E885 Nbg-M |
|     | SICOMP M70                    | E80850-A1-X-X      |
|     | SICOMP M76                    | Anfrage E885 Nbg-M |
|     | SICOMP M80                    | Anfrage E885 Nbg-M |

|      |                                |                    |
|------|--------------------------------|--------------------|
| /10/ | SICOMP M Bedienungsanleitungen |                    |
|      | SICOMP M20                     | E80850-A149-X-X    |
|      | SICOMP M25                     | Anfrage E885 Nbg-M |
|      | SICOMP M26                     | Anfrage E885 Nbg-M |
|      | SICOMP M30                     | E80850-A135-X-X    |
|      | SICOMP M50                     | Anfrage E885 Nbg-M |
|      | SICOMP M56                     | Anfrage E885 Nbg-M |
|      | SICOMP M70                     | E80850-A136-X-X    |
|      | SICOMP M76                     | Anfrage E885 Nbg-M |
|      | SICOMP M80                     | Anfrage E885 Nbg-M |

-----  
| Hardware |  
-----

Beschreibungen der Zentraleinheiten SICOMP M

|      |       |                    |
|------|-------|--------------------|
| /25/ | ZE 01 | E80520-T56-X-X     |
| /26/ | ZE 03 | E80520-T59-X-X     |
| /27/ | ZE 02 | Anfrage E885 Nbg-M |
| /28/ | ZE 04 | Anfrage E885 Nbg-M |

Beschreibungen der Standardperipherie

|      |   |  |
|------|---|--|
| /40/ | Datensichtstation DS 075<br>Technische Beschreibung<br>Betriebsanleitung<br>Bedienungsanleitung | E80520-T110-X-X<br>C79000-38700-C316<br>C79000-M8700-C21 |
| /42/ | Disketten-Laufwerk FD 044   | E85200-T1-X-X  |
| /43/ | Fest-/Wechselplatten-Laufwerke FP 23,<br>PS 048, PS 049   | E80520-T221-X-X  |
| /44/ | Festplatten-Laufwerke FP 24, FP 25  | E80520-T224-X-X  |

---

|      |  |   |
|------|--|---|
| /45/ | Magnetbandkassetten-Laufwerk MK 80/81  | E80850-A3-X-X   |
| /46/ | Magnetband-Laufwerk MB 60  | E80850-A258-X-X   |
| /50/ | Datenerfassungsstationen ES 005, ES 021  | E80520-T8-X-X   |
| /51/ | Drucker DR 202   | E80850-A210-X-X   |
| /55/ | Datenübertragungssteuerung DU 02   | E80520-T222-X-X   |
|      | DU 03  | E80850-A121-X-X   |
|      | DU 05  | E80850-A49-X-X  |
|      | DU 06  | E80520-T201-X-X   |
| /56/ | Datenübertragungssteuerung DU 04   | E80520-T209-X-X   |
| /57/ | Anschaltung KS100, Technische Beschreibung<br>Betriebsanleitung<br>Firmware-Beschreibung               | Anfrage E885 Nbg-M<br>C79000-B8700-C451<br>Anfrage E885 Nbg-M |
| /60/ | PROMEA EA 01<br>PROMEA EA 01, Anschaltmodule ES, DU 04 und MX<br>PROMEA EA 01, Anschaltmodul SINEC TAM | E80520-T60-X-X<br>E80850-A82-X-X<br>Anfrage E885 Nbg-M        |
| /61/ | TREKAM<br>Interne Beschreibung für KS100   | 6BY1404-OXX00-OB  |

Beschreibungen der Prozeßperipherie

|      |   |                  |
|------|---|------------------|
| /85/ | Prozeßeinheit 3600<br>Ein-/Ausgabesteuerung (ES902) | C71000-T8700-C22 |
|      | Prozeßeinheit 3600<br>Prozeßsignalformer (ES 902)   | C71000-T8700-C14 |
|      | Busumsetzer EA 03-U für Prozeßeinheit PE 3600       | E80850-T77-X-X   |
| /86/ | Prozeßeinheit PE F7                                 | E80850-A83-X-X   |
|      | Busanpassung EA03-A und EA02                        | E80850-A5-X-X    |

---

| Systemsoftware |

---

Beschreibungen der Grundprogramme / Organisationsprogramm

|       |                              |                  |
|-------|------------------------------|------------------|
| /100/ | ORG-M<br>Kurzbeschreibung    | 6BB9300-OXX00-OC |
| /101/ | ORG-M<br>Programmierhandbuch | 6BB9300-OXX00-OG |

---

|       |   |                     |
|-------|---|---------------------|
| /102/ | ORG-M<br>Inbetriebnahmeanleitung                            | 6BB9300-OXX00-OK    |
| /103/ | ORG-M<br>Bedienungshandbuch                                 | 6BB9300-OXX00-OA    |
| /105/ | Prozeß- und Sonderverkehr ORG SOPEM<br>Beschreibung         | 6BB2018-OXX00-OB    |
| /106/ | Sonder- und Service-Aufrufe ORG SOAM<br>Programmierhandbuch | 6BB2013-OXX00-OG    |
| /107/ | PSDCACHE, Beschreibung                                      | 6BB9301-OXX00-OB    |
| /108/ | ORG-M<br>Wartungsunterlage                                  | 6BB9300-OXX00-OL    |
| /111/ | ORG 300-PV (ORG des BS 300)                                 | P71100-B0340-X-X-35 |
| /120/ | EVITAM<br>Beschreibung                                      | 6BY4032-OXX00-OB    |

Beschreibungen der Sprachen und Übersetzungsprogramme

|       |                                |                  |
|-------|--------------------------------|------------------|
| /201/ | ASS-M                          | 6BD2000-OXX00-OB |
| /202/ | ASSM-M                         | 6BD2010-OXX00-OB |
| /205/ | OMAK-M                         | 6BJ1018-OXX00-OB |
| /206/ | MAS-M                          | 6BD4000-OXX00-OB |
| /207/ | MACRO-M                        | 6BD4017-OXX00-OB |
| /208/ | PEARL-MC                       | 6BD3010-OXX00-OB |
| /213/ | LNKR-M                         | 6BD1034-OXX00-OB |
| /216/ | ADAPTADR, Adreßraumerweiterung | 6BD3051-OXX00-OB |

Beschreibungen der Dienstprogramme

|       |         |                  |
|-------|---------|------------------|
| /300/ | INITM   | 6BE9012-OXX00-OB |
| /301/ | COPY-M  | 6BE7056-OXX00-OB |
| /302/ | MCSAVE  | 6BE7100-OXX00-OB |
| /303/ | FILE-M  | 6BE9011-OXX00-OB |
| /304/ | MEDIS-M | 6BE4014-OXX00-OB |
| /306/ | MONIM   | 6BE8008-OXX00-OB |
| /307/ | DEBUG-M | 6BE6032-OXX00-OB |

---

|                                |  |  |
|--------------------------------|--|--|
| /308/                          | LIDAM  | 6BY4020-OXX00-OB   |
| /309/                          | TRACE  | 6BY4015-OXX00-OB   |
| /310/                          | TSMM, Kundenbeschreibung   | 6BY4002-OXX00-OB   |
| /311/                          | MBDO-M   | 6BB6037-OXX00-OB   |
| /312/                          | TAPE-M   | 6BE7083-OXX00-OB   |
| /313/                          | TEST-M   | 6BE6021-OXX00-OB   |
| /314/                          | TESTS-M  | 6BE6070-OXX00-OB   |
| /315/                          | LDFIRM   | 6BE7028-OXX00-OB   |
| /319/                          | BIBEAS-M   | 6BE3008-OXX00-OB   |
| /323/                          | FWV, Firmwareverwaltung  | 6BB1400-OXX00-OB   |
| <u>Sonstige Beschreibungen</u> |  |  |
| /400/                          | Speicherzugriffssystem CAGE-M  | 6BB6110-OXX00-OB   |
| /401/                          | DVS-M<br>Beschreibung<br>Bedienungsanleitung   | 6BB6065-OXX00-OB<br>6BB6065-OXX00-OA   |
| /402/                          | SPOOL-M  | 6BB2288-OXX00-OB   |
| /403/                          | SINEC-M  | 6BB4030-OXX00-OB   |
| /404/                          | LINK-M   | 6BB9100-OXX00-OB   |
| /453/                          | SERVICYSYSTEM<br>Kurzbeschreibung<br>Serviceprogramme, Beschreibung<br>Programmieranleitung<br>Bedienungsanleitung                 | 6BY4400-OXX00-OC<br>6BY4400-OXX00-OB<br>6BY4400-OXX00-OG<br>6BY4400-OXX00-OA |
| /500/                          | Betriebssystem BS3/BS4:<br>Bedienungsanleitung<br>Beschreibung (Dienstfunktionen)<br>Programmieranleitung<br>Administratorhandbuch | 6BB9430-OEX00-OA<br>6BB9430-OEX00-OB<br>6BB9430-OEX00-OG<br>6BB9430-OEX00-OE |
| /501/                          | AMBOSS-Generator AMGENM<br>Beschreibung  | 6BE5430-OXX00-OB   |
| /502/                          | Systembeschreibung AMBOSS<br>Bedienungsanleitung AMBOSS 4<br>Projektierungshandbuch AMBOSS 4                                       | 6BA9445-OEX00-OC<br>6BA9445-OEX00-OA<br>6BA9445-OEX00-OH                     |

## 1 Einführung

ORG-Aufbau und -Aufgaben      Leistungsumfang und Arbeitsgeschwindigkeit moderner Büro-computer und Prozeßrechneranlagen lassen sich nur dann wirtschaftlich ausnutzen, wenn

- das Computersystem die organisatorischen und koordinierenden Aufgaben seines Betriebes weitgehend selbst übernimmt,
- die Abläufe der Schaltnetzwerke (Hardware) durch programmierte Funktionen erweitert werden und
- solche durch Programme (Software) realisierten Eigenschaften der Anlage zentral verfügbar sind.

Die Erfüllung dieser Forderungen ist Aufgabe des Organisationsprogramms (ORG), das damit zu einem integrierten Bestandteil des Computersystems wird.

Hauptteile des ORG      Funktionen und Bausteine des ORG lassen sich entsprechend ihren Aufgabenbereichen nach folgenden Hauptteilen gliedern:

- Zentralteil für die Verwaltung von Speicherbereichen, Listen, Warteschlangen und internen Nahtstellen sowie für die Bearbeitung von Fehlerfällen, Anforderungen, Simulationen, Zeitfunktionen und Wiederanlaufmaßnahmen.
- Programmorganisation für Hauptspeicherresidente Programme (HRP) und peripherenspeicherresidente Programme (PRP) zur Bearbeitung der Programmwechsel, Zustandswechsel, Platzwechsel und Koordinierungsaufgaben. Laufbereichsverwaltung.
- Ein-Ausgabe-Organisation zur Koordinierung des Verkehrs der Programme mit den peripheren Geräten.
- Dateiorganisation.
- ORG-nahe Programme für das Generieren, Laden, Bedienen
- Hilfsfunktionen zum Konvertieren und für System-Informationen.

Das ORG erhält die Anstöße zur Erledigung seiner Aufgaben durch folgende Ereignisse, die jederzeit eintreffen können:

- Programmlaufbesonderheiten
- Anforderungen peripherer Geräte (s. 5)
- Aufrufe der Programme (s. 3)

Prioritätsstruktur      In der Hardware-Struktur des Zentralprozessors sind verschiedene Ebenen, Zustände und Prioritäten zu unterscheiden.

Der Zentralprozessor ist mit 16 Ebenen mit je zwei Modi, Normalmodus und Simulationsmodus, ausgestattet. Daraus ergeben sich 32 Zentralprozessorzustände. Jede Ebene besitzt mindestens eine eigene Parametertafel.

Bild 1/1 zeigt die Hardwareebenen für SICOMP M30 bis M80 im Normalmodus. Im Simulationsmodus arbeiten das ORG und ggf. Anwendersimulationen. Geringfügige Differenzen bei SICOMP M20 bis M26: die Funktionen der Ebene 1 kommen nach 2 und die von 0 nach 1; Ebene 0 steht für die Stütztreiber zur Verfügung (s. /9/).

Die Festwertspeicherprogramme Anlagen-STOP, virtuelle Konsole, Urlader, Spannungsausfall und Fehlerbehandlung nehmen eine Sonderstellung ein. Sie werden im Zentralprozessorzustand ZØ mit höchster Priorität abgewickelt, sind keiner Ebene zugeordnet und besitzen keine Parametertafel.

Der Anwender benutzt die Bedienung im STOP-Zustand (virtuelle Konsole)

- bei Anlauf, Wiederanlauf und Neustart
- bei Test- und Prüfarbeiten
- im Störfall.

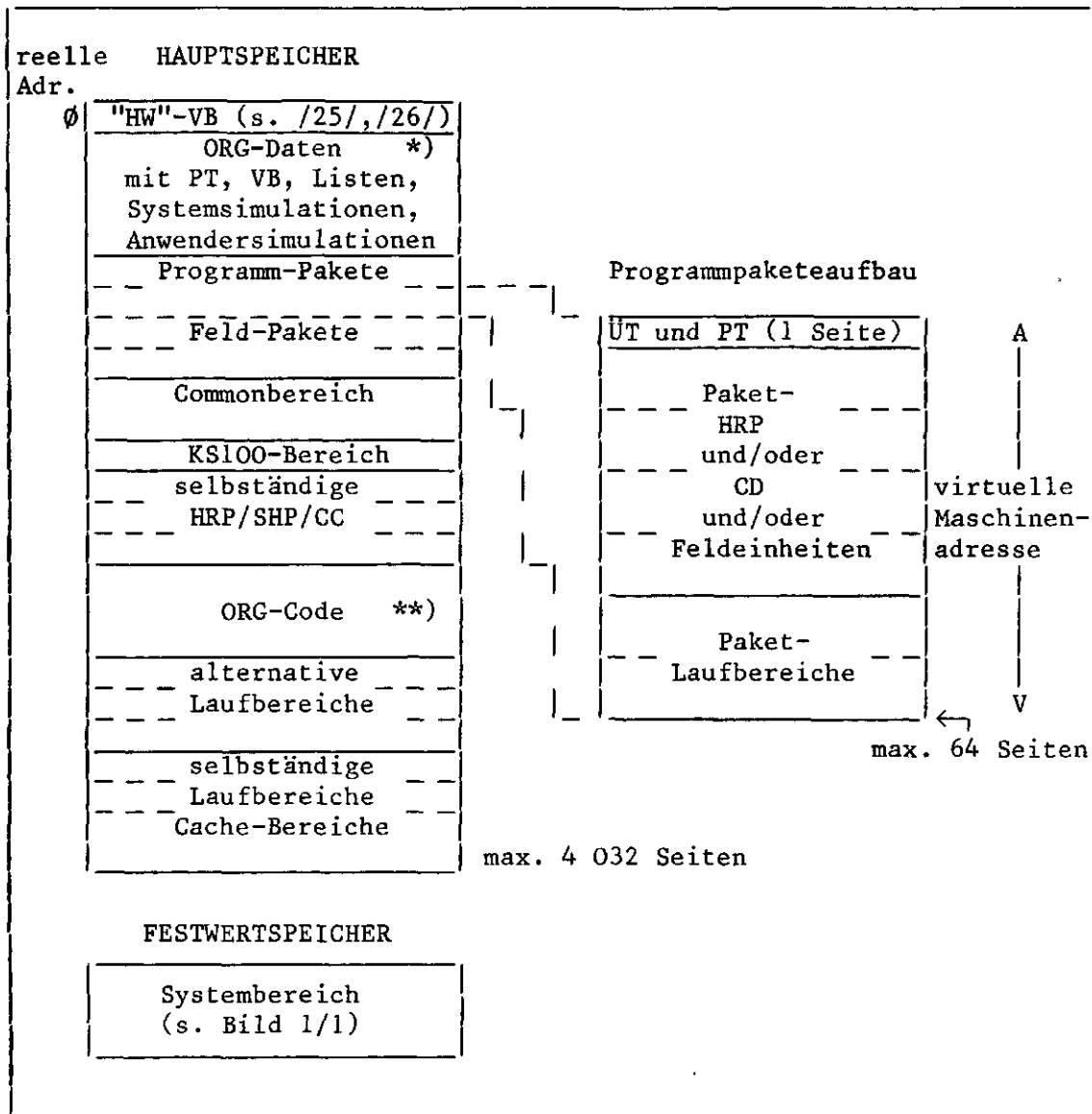
Das ORG nimmt gegenüber allen anderen Programmen eine Ausnahmestellung ein:

- Es wird über Hardware-Funktionen identifiziert,
- kann auf allen Ebenen arbeiten und
- bedient sich mehrerer Parametertafeln.

| Ebenen-Nummer | Priorität  | Funktion   | Zustand                                      | Verwendung                          |  |
|---------------|--|--|--|-------------------------------------|--|
| -             | höchste  | Anlagen-STOP, virt. Konsole, Urlader, Spannungsausfall, Wartungsprogr. | Z0   | Festwertspeicherprogramm            |  |
| 0             | A<br> <br> <br> <br> <br> <br> <br> <br> <br> <br> <br> <br> <br> <br> <br> <br>niedrigste | Fehlerbehandlung u.ä.  | 0  | ORG                                 |  |
| 1             |  | E/A-Verkehr  | 1  | Service-Treiber<br>Anwender-Treiber |  |
| 2             |  |  | 2  | PSD-Treiber                         |  |
| 3             |  | Aufrufbearbeitung, E/A-Verkehr mit Standardperipherie                  | 3  | ORG                                 |  |
| 4             |  | Test und Wartung   | 4  | Service, Wartungsmittel             |  |
| 5             |  |  | 5  |                                     |  |
| .             |  |  | .  |                                     |  |
| .             |  |  | .  |                                     |  |
| .             |  |  | .  |                                     |  |
| .             |  |  | .  |                                     |  |
| 15            |  |  | Standard-Echtzeit-Bearbeitung, Dialogbetrieb | .                                   | Anwenderprogramme, Lader, Monitor, SBP |
|               |  |  |  | 15                                  |  |

Bild 1/1: Hardware-Ebenen (Nummer und Priorität) des Zentralprozessors und ihre Funktionen im Normalmodus





\*) können (je nach Generierangabe) auf 1, 2 oder 3 ORG-Datenadrese Räume verteilt werden (siehe Adreßraumerweiterung, 1.1)

\*\*\*) kann (je nach Generierangabe) auf 1, 2 oder 3 ORG-Codeadrese Räume verteilt werden (siehe Adreßraumerweiterung, 1.1)

Bild 1/2: Speicherplatzaufteilung des Hauptspeichers

## 1.1 Interne Funktionen, Hauptspeicher, Adressierung

**Interne Funktionen**      Soweit interne, zentrale Funktionen nicht in den späteren Hauptabschnitten beschrieben sind, dienen sie im wesentlichen ORG-internen Aufgaben, deren Kenntnis für den Anwender in der Regel entbehrlich aber fallweise für das Verständnis der Zusammenhänge auch nützlich ist.

- . **Listenpool**      Zur Platzersparnis decken Warteschlangen und dynamische Listen des ORG ihren Bedarf aus einem Listenpool, dessen Länge der Anwender beim Generieren abschätzt und festlegt.
- . **Verständigungsbereich (VB)**      Im Verständigungsbereich am Anfang des Hauptspeichers legt das ORG die Anfangsadressen der Listen sowie weitere Hilfszellen, Konstanten, Textteile und sonstige Daten ab, die zentral zugänglich sein müssen.
- . **Segmentierung**      Zur Platzersparnis im Hauptspeicher sind ORG-Bausteine für weniger zeitkritische Funktionen auf dem peripheren Speicher abgelegt (Segmente). Sie werden im Bedarfsfall selbsttätig in ORG-eigene Hauptspeicher-Laufbereiche (zwei Segmentpuffer im VB) übertragen.

Der Benutzer muß Auswirkungen auf die zeitliche Koordinierung bedenken, soweit seine Programme segmentierte Funktionen ansprechen.

Auf Wunsch können beim Generieren die ORG-Bausteine auch im HSP resident abgelegt werden.

- . **Bausteinablauffolge**      Die von den Programmen an das ORG gerichteten Anforderungen (Aufrufe) werden in Aufträge unterteilt, die in Auftragswarteschlangen geführt nacheinander zur Ausführung kommen (Auftragsbearbeitung).

**Speicherplatzaufteilung**      Bild 1/2 zeigt die Aufteilung des Hauptspeichers, der Programme und Daten aufnehmen kann. Er ist eingeteilt in Seiten; bis zu 4032 Seiten kann er umfassen.

[ 1 Seite = 1024 Wörter = 2048 byte ]

Die Datenbereiche des ORG, insbesondere Hardware-Verständigungsbereich und Simulationen, belegen den ersten Teil des HSP, den reellen Adreßraum mit den Seiten 0 bis max. 64; diese Seiten werden reell adressiert. Bei Adreßraumerweiterung (s.u.) stehen noch ein oder zwei weitere Adreßräume für ORG-Daten zur Verfügung.

Die Codebereiche des ORG stehen in einem Block vor den Laufbereichen: ein bis drei Adreßräume von je max. 64 Seiten, adressiert über den Tafelzeiger TZ1 (bei ZE01) bzw. TZ3 (bei ZE02, ZE03, ZE04).

Die ORG-nahen Programme laufen in den Laufbereichen ab.

Weitere HSP-Blöcke (für Pakete, Commonbereich, selbständige Objekte, selbständige Laufbereiche) stehen den entsprechenden Anwender-Objekten zur Verfügung. Ggf. existieren noch die Blöcke KS100-Bereich und Cachebereiche.

**Adreßräume** Der Adressierungsmechanismus des Systems kann einen HSP-Bereich bis zu 64 Seiten ansprechen. Bei der virtuellen Adressierung sind die ansprechbaren Seiten in der Übersetzungstafel eingetragen. Ein Adreßraum ist der in seiner Übersetzungstafel beschriebene Bereich. So kann man sich den HSP in Adreßräume eingeteilt denken. Die Einteilung erfolgt beim Generieren des ORG (Adreßräume des ORG, der Pakete, des Commonbereichs, der selbständigen Laufbereiche u. a.) und beim Laden von selbständigen Objekten. Adreßräume lassen sich im Betrieb manipulieren, müssen keinen zusammenhängenden HSP-Platz belegen und können einander überlappen.

Der reelle Adreßraum kann ohne ÜT erreicht werden; er bleibt dem ORG vorbehalten (s. o.).

**Tafelzeiger** Durch das Einstellen eines Tafelzeigers auf eine bestimmte ÜT kommt der entsprechende virtuelle Adreßraum in den Zugriff eines Programms. Drei Tafelzeiger stehen zur Verfügung.

- . TZ1 Bei der Befehlsbearbeitung werden die Daten/Operanden über Tafelzeiger TZ1 ermittelt (Datenadreßraum).
- . TZ3 Bei der Befehlsbearbeitung werden die Befehlsadressen selbst über den Tafelzeiger TZ3 ermittelt (Codeadreßraum). In der Regel zeigen beide Zeiger TZ1 und TZ3 auf dieselbe ÜT; im selben Adreßraum befinden sich Daten und Code. In besonderen Fällen kann der TZ3 umgestellt, und so können Befehlsabläufe in einem anderen Adreßraum ausgeführt werden (Adreßraumerweiterung für ORG und Anwenderprogramme, s. u.). Die ZE01 besitzt keinen TZ3; der TZ1 übernimmt die Ermittlung von Befehls- und Datenadressen.
- . TZ2 Der Tafelzeiger TZ2 erlaubt, auf Daten in weiteren Adreßräumen zuzugreifen (s. 4.7).

**Adreßraum-  
erweiterung** Da bei umfangreichen Systemen die beiden ursprünglichen Adreßräume für das ORG nicht mehr ausreichen (d.h. ein entsprechender Generiervorgang würde dann wegen unzulässig großer ORG-Objekte abgebrochen werden), ermöglicht eine bestimmte Generierangabe die Belegung von 4 bzw. 6 Adreßräumen zu je max. 64 Seiten durch das ORG. Dies ist bei allen Anlagentypen ausgenommen SICOMP M20 und M30 (ZE01 /26/) möglich.

Zum ORG gehörige Datenbereiche bestimmter "ORG-Subsysteme" werden aus dem realen Adreßraum (ersten Datenadreßraum) herausgenommen und in einen neuen, den zweiten (bzw. dritten) Datenadreßraum, verlegt. Zur Adressierung benutzt das ORG die Tafelzeiger TZ1 bzw. TZ2 und die "Daten-Übersetzungstafel".

Außerdem werden zum ORG gehörige Codebereiche bestimmter "ORG-Subsysteme" aus dem ersten Codeadreibraum herausgenommen und in einen neuen, den zweiten (bzw. dritten) Codeadreibraum, verlegt. Zur Adressierung der drei Codeadreibräume benutzt das ORG den Tafelzeiger TZ3 und je eine eigene "Code-Übersetzungstafel".

Der zweite Codeadreibraum sowie der zweite Datenadreibraum des ORG können für folgende Subsysteme genutzt werden: SPOOL-M, DVS-M, CAGE-M (enthält PAGE-Baustein), SNFV-M, LANRFA und MRTS.

Der dritte Codeadreibraum sowie der dritte Datenadreibraum des ORG können für die Treiberbausteine zur Datenübertragungseinheit KS100, für den ausgelagerten PSD-Cache sowie für das Subsystem LANRFA genutzt werden.

Adreibraum-  
erweiterung,  
Anwender

Ohne besondere Maßnahmen ist ein Programm in seiner Länge begrenzt; es muß in einen Adreibraum passen (Lader und Binder verkraften nur Objekte bis zu einer Länge von 65409 Wörtern (64 Seiten minus 128 Wörter für ÜT/PT). In folgenden Fällen kann diese Begrenzung umgangen werden:

- HSP-segmentierte Programme (SHP); die ÜT wird ständig umgeladen (s. 2.1 und 4.9).
- Auslagern von Daten in fremde Adreibräume; Zugriff über TZ2 (s. 4.7).
- Auslagern von Codeteilen (Unterprogrammen) von Subsystemen in einen fremden Adreibraum, die dann durch Einstellen des TZ3 angesprungen werden (s. /216/).

Speicherschutz Mit Ausnahme der reellen Adressierung, die keinen Schutz ermöglicht, aber für Anwenderobjekte nicht vorgesehen ist, erfolgen alle HSP-Zugriffe mittels Übersetzungstafeln (ÜT). Damit sind zwangsläufig für jedes Objekt nur solche Seiten erreichbar, die mit der gleichen ÜT innerhalb eines 64-Seiten-Bereichs arbeiten und für die in der ÜT ein Eintrag vorliegt. Alle übrigen Speicherbereiche sind ohne besondere Maßnahmen vom betrachteten Objekt aus nicht zugänglich, also geschützt.

Adressierung

. Adreib-  
volumen

Assembler und Grundsprache stellen für Befehls- und Operandenadressen im HSP 16 bit zur Verfügung. Damit ist ein Adreibvolumen von 64 Seiten beschreibbar. Dies wird auch als "virtueller Adressierungsbereich" bezeichnet, in den - zunächst ohne Bezug auf einen physikalischen Speicherort - die Grundsprache ein übersetztes Programm abbildet (Grundsprache-Adressen, Übersetzungsprotokoll).

---

Ob das Adreßvolumen des Übersetzers (und des virtuellen Adressierungsbereichs) größer oder kleiner als der reelle (physikalische) Speicher ist, in dem das Programm ablaufen soll, ist dabei grundsätzlich ohne Belang. Der Lader erzeugt aus den Grundspracheadressen die *v i r t u e l l e n* Adressen von 16 bit im virtuellen Speicher abhängig von Objektart und Speicherbereich. Der virtuelle Speicher von max. 64 Seiten läßt sich als Bereich des HSP verstehen (Beispiele: Laufbereich, Commonbereich, Paket).

- . virtuelle Adressierung    Es sind nunmehr Gesetzmäßigkeiten und Einrichtungen nötig, die die Zuordnung der virtuellen Adresse zu einer reellen HSP-Adresse regeln und vornehmen. Das können Hardware-Funktionen, Software-Verfahren oder Kombinationen beider sein (Adreß-Übersetzungsvorgang mit Tafelzeiger, Übersetzungstafeln, s. /25,26/). So sind für alle Anwenderobjekte die virtuellen Adressen von 16 bit in 22 bit lange reelle Adressen umzuwandeln.

✓

✓

✓

✓

1.2 Leistungsmerkmale

1.2.1 Struktur- und Betriebsparameter

|                                   |  |
|-----------------------------------|--|
| PRODUKTART,<br>VERWENDUNG         | Prozeßrechner-Realzeitbetriebssystem (BS-M)<br>Bürocomputer-Dialogbetriebssystem (AMBOSS 4),<br>Peripheralspeicher-ORG / Hauptspeicher-ORG   |
| RESIDENZ,<br>LAGE,<br>PLATZBEDARF | : zwei bzw. vier bzw. sechs HSP-Teile (Code, Daten)<br>wählbaren Umfangs bis je 64 Seiten<br>. ab reellem Speicheranfang die Daten und Listen ein-<br>schließlich der ORG-eigenen Segment-Wechselbereiche<br>(2 mal 1536 Wörter)<br>. im virtuellen Bereich der ORG-Code sowie - bei<br>Adreßraumerweiterung - weitere ORG-Daten<br><br>: Standard-Laufbereich(e) im HSP u.a. für Lader,<br>Standardbedienprogramm und Monitor<br><br>: Segmente und Systemdateien in ORG-eigenen PSD-Bereichen<br>je nach Systemausstattung   |
| STRUKTUR                          | Generierfähiger Masterstapel als Modulbibliothek, feste<br>Grundausrüstung und optionelle Bausteine  |
| EBENENZUORDNUNG                   | Ebene 0-3 : ORG und Treiber<br>Ebene 4 : Service<br>Ebene 5-15: Anwenderprogramme  |
| LIEFERUMFANG,<br>LIEFERFORM       | Lieferung erfolgt grundsätzlich als AMBOSS-Systempaket<br>(sowohl für Betriebssystem BS-M als auch für Betriebs-<br>system AMBOSS 4)<br>: Plattenstapel/Wechselplatte (nur SICOMP M60/70/76/80)<br>: Magnetbandkassette<br>und enthält<br>: als urladefähige Dateien<br>. modellspezifisches Grundbetriebssystem<br>. Testbetriebssystem<br>. anwendungsspezifisches Betriebssystem<br>BS-M oder AMBOSS 4 (auf Bestellung)<br>: in Bibliotheken u.a.<br>. Systemgenerator AMGENM<br>. ORG-Generator MGEN<br>. Dienstprogramm MODGEN<br>. Wartungshilfe EVITAM<br>. ORG-Masterstapel<br>. mindestens eines von sechs Sprachsystemen<br>(nach Bestellung)<br>: weitere Programme siehe /4/ |
| SYSTEMUMGEBUNG                    | : Mindest-HSP-Ausbau 128 Seiten<br>: Mindestausstattung Peripheriegeräte<br>. Plattenlaufwerk  <br>. Datensichtstation > nicht bei HSP-ORG<br>. Magnetbandkassettenlaufwerk  <br>. Zeitgeber   |

-----

GENERIEREN : Betriebssystem BS-M oder AMBOSS 4 ("Knopfdruck-Software")  
entsteht bei Stückprüfung des Systems beim Hersteller,  
bestellbar bei Erstlieferung  
: auf Gast- oder Zielanlage mit  
. Systemgenerator AMGENM mit Bildschirm-Dialog,  
prozedurgesteuert (s. /501/)  
. ORG-Generator, kommandogesteuert  
zur Anpassung an Systemkonfiguration und Anlagenaus-  
stattung

### 1.2.2 Ablauf- und Verwaltungsobjekte

LADEOBJEKTE Programme, Common Data und Common Code; durch Ladevorgänge  
einzubringen

. PROGRAMME selbständig oder in Paketen

Arten : HSP-resident (HRP), Ebene 5 bis 15 } ggf. auch  
: PSD-resident (PRP), Ebene 5 bis 15 } segmentiert  
: Anwendertreiber, Ebene 1 und 2

Anzahl maximal 4094 zusammen mit Common Data und Common Code

Identifikation durch Programmnummern 1 bis 4094 (zusammen mit Common  
Data und Common Code) oder  
durch Programmnamen (max. 6 Buchstaben/Ziffern)

Prioritäten für HRP und PRP 5 bis 15 entsprechend den Ebenen,  
für die Zuteilung von Betriebsmitteln

Länge maximal 65536 Wörter (einschl. ÜT, PT und eingebundener  
Teile), Teile in fremden Adreßräumen sind möglich)

. COMMON DATA immer HSP-resident, V- und I-Teil möglich, aber nicht  
(CD) separierbar; identifiziert durch Objektnummer oder  
Objektnamen, mit Programmen zusammen verwaltet  
: in Paketen (lokal)  
: im Commonbereich (zentral)  
: in Feldpaketen die Feldeinheiten

. COMMON CODE Der Common Code (auch "selbständiger Common Code" genannt)  
(CC) ist immer HSP-resident, I-Teil und mehrfach benutzbarer  
V-Teil, selbständiges Objekt identifiziert durch Objekt-  
nummer oder -namen, mit Programmen zusammen verwaltet

GENERIEROBJEKTE Pakete, Commonbereich, Feldeinheiten und Laufbereiche;  
beim Generieren anzugeben

. PAKETE enthalten mehrere Objekte mit einer Gesamtlänge von  
max. 64 Seiten mit gemeinsamer Übersetzungstafel

Programm- Anzahl: zusammen mit Feld-Paketen max. 15; können HRP,  
Pakete Laufbereiche, Feldeinheiten und CD enthalten, Summe  
der HRP und Laufbereiche je Paket max. 15.



---

|                        |   |
|------------------------|---|
| Feld-Pakete            | Anzahl: zusammen mit Programm-Paketen max. 15<br>Lage: nach den Programm-Paketen (wenn vorhanden), können eine Feldeinheit enthalten.   |
| . COMMONBEREICH        | zur Aufnahme von zentral zu benutzenden Common Data und Feldeinheiten, Länge: max. 63 Seiten  |
| . LAUFBEREICHE         | im virtuell adressierten Speicherbereich, Gesamtanzahl maximal 255, durch Laufbereichsnummern 1..255 bezeichnet   |
| selbständige           | Länge maximal 64 Seiten (einschl. ÜT und PT);<br>PRP beim Laden fest zugeordnet   |
| in Paketen             | Länge maximal 63 Seiten<br>Lage am Paketende bündig abschließend;<br>PRP beim Laden fest zugeordnet   |
| alternative            | eine Gruppe von selbständigen Laufbereichen, die dynamisch den PRP zugeordnet werden (optimierend beim PRP-Eintransfer); nur die ganze Gruppe wird identifiziert.   |
| Standard-Laufbereich   | als solcher wird ein Alternativlaufbereich oder, wenn keiner generiert, der Laufbereich mit der Nummer 1 erklärt; Länge mindestens 20 Seiten  |
| . CACHE-BEREICHE       | Am Speicherende können max. 8 Cachebereiche für die PSD-Cache-Funktion stehen. Die Länge ist praktisch davon abhängig, wieviel Platz der Anwender für die Cachefunktion im HSP zur Verfügung stellen will.                                      |
| . SIMULATIONS-ROUTINEN | immer HSP-resident, Ablauf von Hardware gesteuert.<br>: Systemsimulationen als Bestandteil des Masterstapels, generierbar, im Adreßbereich des ORG<br>: Anwendersimulationen vom Anwender in Masterstapel einzubringen, maximale Anzahl ca. 250 |
| . FELDEINHEIT          | zur dynamischen Verwaltung von global benutzbaren Daten-Feldern<br>Struktur: statisch oder dynamisch definierbar<br>Typen: im Programmpaket, im Commonbereich, im Feldpaket<br>Lage und Länge: typabhängig<br>Anzahl: max. 15                   |
| WARTEBEREICH           | PSD-Bereich für alle ruhenden oder wartenden PRP; eine Datei, deren Länge beim Generieren anzugeben ist.  |
| DATEIEN                | strukturierte Datenmengen, den logischen Datenträgern (Geräten) zugeordnet  |
| Arten                  | : Systemdateien (vom Generator selbsttätig eingerichtet)<br>: Anwenderdateien (vom Benutzer einzurichten)<br>. Prozeßdateien: schneller Zugriff durch HSP-residentes Buchführungselement  |
| Lebensdauer            | bleiben über Wiederanlauf hinweg erhalten   |
| Speichermedien         | : Platte<br>: Diskette  |

---

|                |   |
|----------------|---|
| Datenarten     | alphanumerisch oder binär   |
| Anzahl         | je logischen Datenträger<br>max. 7808 bei Platten, max. 976 bei Disketten                   |
| Länge          | bis 65536 Sektoren je Datei, begrenzt durch den logischen Datenträger (1 Sektor = 512 byte) |
| Identifikation | durch Namen aus 6 Zeichen, vom Benutzer festzulegen   |

SPEICHERPLATZ-  
VERWALTUNG/VERGABE

- . im Hauptspeicher (s. Bild 1/2) in aufsteigender Reihenfolge vom Anfang belegt durch:
  - : ORG-Daten-Bereich: durch Generator festgelegt
  - : Pakete
    - . Längen nach Anwendergenerierangaben
    - . Lage aufsteigend vom Generator festgelegt
    - . Platzaufteilung in Paketen nach Ladeparametern
  - : Commonbereich
    - . Länge und Schutzgrenze nach Anwendergenerierangaben
    - . Lage anschließend an Pakete
    - . Platzaufteilung im Commonbereich nach Ladeparametern
  - : KS100-Bereich
  - : Bereich für selbständige HRP, SHP und CC
    - . Platzvergabe durch ORG in Vielfachen von Seiten
  - : ORG-Code
    - . Länge bestimmt durch Anwendergenerierangaben
    - . Lage vor den Laufbereichen
  - : Laufbereiche, alternative, danach selbständige
    - . Längen nach Anwendergenerierangaben
    - . Lage vor den Cachebereichen
  - : Cache-Bereiche
    - . Länge bestimmt durch Anwendergenerierangaben
    - . Lage ab Speicherende in Richtung fallender Adressen durch Generator bestimmt
- . im Peripherenspeicher
  - : Platzverwaltung auf Dateibasis wahlweise
    - . durch ORG (fehlende Anwenderangaben)
    - . durch Anwendervorgabe beim Einrichten
- SOFTWARE-  
BETRIEBSMITTEL
  - : im ORG-Masterstapel enthalten:
    - . Lader (2 Ausführungen)
    - . Standardbedienprogramm (2 Ausführungen)
    - . Monitor im BS-M
    - . Zusatz- u. Hilfsfunktionen (s. 1.2.7)
  - : vom Anwender einzubringen:
    - . Wiederanlaufprogramm

GERÄTE

- . Identifikation über beim Generieren vereinbarte und der E/A-Adresse zugeordnete logische Gerätenamen
  
- . Arten
  - Standardgeräte : Ein- und/oder Ausgabegeräte
  - : je nach Medium
    - . mit Papierdatenträger
    - . mit magnetischem Datenträger
    - . ohne Datenträger (z.B. Datensichtstation, Datenübertragungsperipherie, Teleservice-Anschaltmodule)
  - : ladbare (mit Firmware) und nicht ladbare Geräte und Anschaltungen
  
- Prozeßgeräte : alarmbildende Prozeßsignalformer
- : Signalformer ohne zugeordnete ORG-Bausteine
  
- Sonderperipherie-Geräte : Geräte mit Anwendertreiber, Betrieb nur bedingt ORG-unterstützt

ORG-INTERNE  
OBJEKTE

- . Verständigungsbereich (VB)
- . Pool für Listen und Warteschlangen und Dateibuchführung
- . ORG-Segmentbereiche
- . Programmnummernliste
- . Geräteliste
- . Laufbereichsliste
- . Paketliste
- . Gerätepuffer/Pufferzone bei Datenpufferung

EINBRINGEN VON  
OBJEKTEN

- : beim Generieren (Pakete, Bereiche, Geräte, Anwendersimulationen)
- : durch Laden (Ablaufobjekte, Anwendertreiber, Daten)
- : durch Einrichten (Dateien)

1.2.3 Funktionen

Betriebsabwicklung, Zuordnungen, Datenaustausch, Dateiverkehr, Programmablauf, Koordinierung

ORG-ANSTOSS

- ereignisgesteuert durch:
- . ORG-Aufrufe
  - . Hardwaresignale (Interrupts)
  - . Fehler

NAHTSTELLEN ZUM  
BENUTZER

- . Aufrufe aus Anwenderprogrammen
- . Kommandos an Standardbedienprogramm
- . Kommandos an Bediensystem (AMBOSS 4, s. /500/)
- . Meldungsausgaben des ORG an den Bediener
- . Quittungseingaben des Bedieners an das ORG
- . Anzeigen des ORG in Aufrufen der Programme
- . Objektparameter- und Zustandsmitteilungen des ORG (in Verbindung mit Aufrufen und Kommandos)

- 
- BETRIEBSMITTEL-  
ZUTEILUNG** erfolgt durch das System statisch für  
: den Speicherplatz
- oder dynamisch mittels nach Zeit oder Priorität ab-  
gearbeiteter Warteschlangen für  
: den Zentralprozessor (Multiprogramming)  
: die Geräte (ggf. in Verbindung mit Datenpufferung)  
: die Laufbereiche
- LAUFBEREICHS-  
ZUORDNUNG DER PRP** : fest, beim Laden der Programme vom Benutzer ange-  
geben  
: dynamisch beim Eintransfer der Programme für Alter-  
nativlaufbereiche
- PLATZWECHSEL  
DER PRP** Räumung eines Laufbereiches nur bei Bedarf und auto-  
matische spätere Programmfortsetzung,  
: je nach Priorität, wenn Räumbarkeit gegeben:  
. unbedingt, streng nach Priorität  
. bedingt, an Wartestellen  
: ferner sind möglich:  
. Räumung per Aufruf (Bereichsverzicht)  
. Platzwechselsperre (Räumungsverbot)  
. Platzwechsel nach Zeitscheibe
- SEGMENTIERTE  
PROGRAMME** ORG-unterstützt. Nach dem Binden der Moduln mit Struk-  
turfestlegung: Erstellen der Arbeitsform mit Einbau  
der Segmentwechselroutine (Grundsprache-Lader). Möglich  
sind  
. segmentierte HRP und PRP mit Segmenten auf PSD  
. HSP-segmentierte Programme (SHP)  
mit Segmenten im HSP
- DATENAUSTAUSCH  
ZWISCHEN PROGRAMMEN** je nach Datenmenge, Objektort und Geschwindigkeit:  
. über max. 2 Register (HSP)  
. mittels Feldübertragung (HSP und PSD)  
. mit Common Data (HSP)  
. mit dynamischer Datenfeldverwaltung (Feldeinheiten)  
. über Dateien (PSD)  
. programmierte Bedienungsübergabe
- DATEIVERKEHR**
- . organisatorische Funktionen
    - . einrichten und löschen
    - . eröffnen und schließen
    - . Zeiger lesen, einstellen
    - . Etikett lesen, schreiben
    - . Buchführung lesen
    - . Dateilänge ändern
  - . Zugriff/Transfer : Lesen oder Schreiben
    - alphanumerisch oder binär
      - . seriell (satzweise) oder direkt mit Adreßangabe
  - . Schutz : Schreibschutz: je Datei für alle Benutzer
    - : Dateischutz durch
      - . Eigentümerkennzeichen:  
eingeschränkte Änderbarkeit der Dateieigenschaften
      - . Benutzerkennzeichen: eingeschränkter Benutzerkreis

- 
- OBJEKTAKTIVIERUNG erfolgt durch
- . Aufrufe
  - . Kommandos
  - . periphere Anforderungen (Hardware-Interrupts)
  - . Weckerfunktionen (Zeitablauf)
- PROGRAMMABLAUF-  
STEUERUNG
- : durch Aufrufe, teilweise auch Kommandos für:
    - . Starten
    - . Beenden (eigenes Programm)
    - . Beenden (fremdes Programm)
    - . Anhalten
    - . Fortsetzen } zwei Aufrufpaare
    - . Startsperr setzen/löschen
    - . Wartefunktionen
    - . Platz/Programmwechselfunktionen
    - . Koordinierungszähler (Semaphorvariable) universell  
verwendbar, auch mit Datenübergabe
    - . Testfunktionen
  - : durch Zeitparameter in Aufrufen
  - : durch Monitorfunktionen (Stapelverarbeitung)
  - : periphere Anforderungen
- SIMULTANARBEIT  
(Multiprogramming  
und Geräteverkehr)
- : ablauffähige Programme im HSP erhalten den Zentral-  
prozessor nach Priorität, bei gleicher Priorität  
nach Zeit, zugeteilt
  - : ORG koordiniert die Gerätezugriffe der Programme  
durch
    - . Führung gerätespezifischer Transfer-Warteschlangen
    - . Bearbeitung von Wartefunktionen  
(implizites, explizites, Mehrfach-Warten)
    - . Programmfortsetzung nach Operationsabschluß
  - : Datenpufferung erhöht den Grad der Simultanarbeit  
zwischen Programmen und Geräten
- INFORMATIONEN  
ÜBER OBJEKZUSTÄNDE
- sind erhältlich
  - : implizit durch Anzeigen oder Parameter in Aufrufen  
mit Objektzugriff oder durch Meldungen
  - : explizit durch Aufrufe zur Zustandsermittlung
  - : mit Kommandos für Objekt- und Systemparameter
- KOORDINIERUNG  
DER ZUGRIFFE ZU  
OBJEKTEN
- : über Warteschlangen des ORG
  - : mittels Koordinierungszähler für Anwenderprogramme
  - : siehe Programmablaufsteuerung
- KOMMUNIKATION
- : zwischen Rechneranlagen über Aufrufe an Datenüber-  
tragungsperipherie
  - : zwischen Geräten über Peripheralspeicher-Transfer
  - : zwischen Programmen durch
    - . Datenaustausch (s.o.)
    - . programmierte Bedienung
  - : zwischen Bediener und Programmen mittels Kommando-  
eingaben und Meldungsausgaben
    - . Funktionen des Bediensystems (s. /500/)

---

#### 1.2.4 Unterbrechungssystem

- UNTERBRECHUNGS-      Anzahl 16, je Ebene zwei Zentralprozessorzustände:  
EBENEN                Normal- und Simulationsmodus (s. Bild 1/1)
- UNTERBRECHUNGS-      : Hardware:  
EREIGNISSE            . Geräteanforderungen  
                         . Netzausfallfrühwarnung, Netzausfall  
                         . Eingabe von virtueller Konsole
- : Software  
                         . Programmfehler  
                         . ORG-Aufrufe  
                         . Prozeßsteuerbefehle
- REAKTIONSZEIT      : wenn keine höherprioren Hardware-Anforderungen vor-  
BEI UNTERBRECHUNGS-      liegen bei Programmlaufbesonderheiten  
ANFORDERUNG            . in Ebene 1 und 2: zweimal ZWZ \*)  
                         . in Ebene 5 bis 15: einmal ZWZ \*)
- : abhängig von gleich- oder höherprioren Hardware-  
                         Anforderungen
- \*) ZWZ = Zustandswechselzeit der Hardware (ohne Soft-  
                         warelaufzeiten)
- ALARMPROGRAMME      : Struktur:  
s. /105/                . Programm Anmeldung (Aufruf) für zugeordnete Sig-  
                         nalformer bzw. E/A-Adresse beim ORG  
                         . Warten auf Alarm  
                         . Programmfortsetzung durch ORG bei eingetroffenem  
                         Alarm  
                         . Einlesen und Verarbeiten der Alarmwörter der Sig-  
                         nalformer bzw. der Geräteanforderung durch Pro-  
                         gramm
- : Alarmanzahl je Programm beliebig  
                         : Anzahl alarmverarbeitender Programme beliebig  
                         : Alarmauflösung: abhängig von der Laufzeitsumme in  
                         ORG und Anwenderprogramm, der Alarmanzahl, den An-  
                         schluß- und Programmprioritäten sowie dem Prozeß-  
                         signalformertyp

1.2.5 Ein-Ausgabe-Funktionen

GERÄTE-EIN-AUSGABE- einschließlich der Funktionen für zentrale E/A-Bear-  
BAUSTEINE (TREIBER) beitung im ORG enthalten für:

- . Standard-  
peripherie : Ein/Ausgabeeinheiten
  - . Datensichtstationen
  - . Datenerfassungsstationen
  - . Drucker
  
- : Periphere Speichereinheiten
  - . Wechselplattenlaufwerk
  - . Fest-/Wechselplatte
  - . Festplattenlaufwerke
  - . Diskettenlaufwerke
  - . Magnetbandkassettenlaufwerke
  - . Magnetbandlaufwerke

} (Datei- und  
Geräteverkehr)
  
- : Datenübertragungseinheiten
  - . Datenübertragungssteuerungen
  - . Teleservice-Anschaltmodule
  
- . Netzwerk-  
Kommunikation : Hochleistungsanschlungen für
  - . lokales Netz 'Ethernet'
  
- . Prozeß-  
peripherie : Prozeßsignalformer
  - . Alarmanforderungen
  
- . Sonderperipherie : mittels Anwendertreiber und bedingter ORG-Unter-  
stützung über Standard-E/A-Aufrufe (Online-Betrieb)
  
- GERÄTE OHNE : Betrieb mit Ein-Ausgabe-Maschinenbefehlen und Makro-  
ZUGEORDNETE aufrufen für  
ORG-BAUSTEINE
  - . Digital- und Analog-Eingabe und -Ausgabe
  - . integrierende Analogeingabe
  - . Momentanwertanalogeingabe
  
- : Anwendertreiber ohne ORG-Unterstützung mit Maschinen-  
befehlen (Offline-Betrieb)
  
- VERKEHRSABLAUF Simultananarbeit unabhängiger Geräte untereinander und  
zwischen Programmen und Geräten möglich
  
- . Anstoß durch geräteunabhängige Standardaufrufe } an das ORG  
oder Spezialaufrufe
  
- . Koordinierung : Gerätezugriffe
  - . prioritätsgeordnete Warteschlangen je Gerät/Teil-  
funktion im ORG (Aufrufbearbeitung)
  - . Zugriffskoordination durch Koordinierungszähler  
(Anwenderfunktion)
  
- : Programmabläufe
  - . Funktionen zum Warten auf Operationsende
  - . Mehrfachwarten und Sonderfälle mittels Koordinie-  
rungszähler
  
- : Geräte online/offline schalten

- 
- . Gerätebetrieb besteht an E/A (= Transfers) und Steueroperationen, Funktionsumfang für Standardperipherie
    - . fest (Standard)
    - . variabel durch ladbare Firmware, alternative Gerätefunktionen (FWV, /323/).
  
  - . Transfer : hardwareseitig, je nach Gerät
    - . Blockverkehr
    - . Zeichenweiser Verkehr  
: softwareseitig je nach Datenart und Aufrufparametern
    - . logische Sätze (durch Pufferlänge oder Endezeichen bestimmt)
    - . Byte- oder Wortverkehr
  
  - . Daten-Pufferung Aufrufe an (langsame) Standardgeräte mit den Nutzdaten sind pufferbar
    - . für Ein- und Ausgabeoperationen bei PRP, zur Verbesserung des Platzwechselerhaltens (Räumbarkeit), Puffer im HSP
    - . für Ausgabeoperationen bei HRP und PRP, zur Verkürzung der Programmlaufzeit, Puffer auf PSD.
  
  - . Pufferung über PSD-Cache Vorratshaltung von PSD-Daten auf dem HSP (Kacheln im Cache-Bereich) zur Reduzierung der PSD-Zugriffe
  
  - . Zustandsinformationen und Fehlervorkerhungen : Peripherie-Anzeigen in Originalform  
: Aufruf-Anzeigen und Meldungen  
: teilweise Operationswiederholversuche  
: Service-Informationen/-Funktionen
  
  - BEDIENUNG : Bedieneingaben (Kommandos)
    - . manuell, von bedienbaren Geräten
    - . im Steuerkartenbetrieb an das Standardbedienprogramm
    - . als programmierte Bedienübergabe
    - . über Prozeduren an das Bediensystem (s. /500/)  
: Bedieneingaben an
    - . ORG
    - . Standardbedienprogramm
    - . Bediensystem bei AMBOSS 4 (s. /500/)
    - . bedienbare Anwenderprogramme (spezieller Bedieneingabeaufruf, bzw. vom Bediensystem gesteuert)  
: zur Kommandovereinfachung:
    - . Kurzformen durch "Bedienzeichenersatz"  
: Dialogverkehr frei oder  
: Terminalbetrieb (zugeordnet) möglich über
    - . programmierte Aufrufsequenzen (Bedieneingabe-Meldungsausgabe)
    - . Kenntnisgabe des Bedienungsursprungsgerätes bzw. des zugeordneten Terminals



- 
- DATENAUFBEREITUNG erfolgt durch
- . Aufrufparameter
  - . Steuerzeichen
  - . Geräteeigenschaften
- . Datenarten mit zugeordneten Aufrufen für Ein- und Ausgabe von
- : alphanumerischen Zeichenfolgen (mit Steuerzeichenerkennung)
  - : Binärdaten (ohne Steuerzeichenerkennung)
  - : Grundsprache (nur Lader mit eigenem Ladeaufruf)
- . Datenfeld-  
adressierung
- : im HSP (intern, eigener und fremder Adreßraum)
    - . direkt oder indirekt mit
    - . Wort- oder Byteadressen
  - : auf PSD (extern) mit
    - . relativen Dateiadressen oder
    - . physikalischen Speicheradressen
- . Codes, Formate
- : Standardcode für die interne Darstellung ist der ISO-7-Bit-Code (DIN 66003)
  - : Ein- und Ausgabeformate nach Anwenderangaben

#### 1.2.6 Sicherungsfunktionen

- FEHLERBEHANDLUNG
- : Melden besonderer Betriebszustände
    - . Peripherie-Anzeigen in Originalform
    - . Meldungen an Textausgabegeräten (Fehlermeldegerät, Standard-Meldegerät, Bedien-/Meldegerät, bedientes Gerät), teilweise mit Quittungsmöglichkeit
  - : bei Übertragungs- oder Bedienungsfehlern  
Aufruf- oder Transferwiederholungen
  - : Abbruch fehllaufender Programme
- SCHÜTZEN VON  
OBJEKTEN
- durch Hardwareeinrichtungen (seitenweise) oder Softwaremaßnahmen in Verbindung mit Adreß- oder Objektzugriff
- . Schreibverbot für die I-Teile der selbständigen Programme durch Bereichsschutz mittels Übersetzungstafeln (nicht ZE01)
  - . Schutz von Dateien gegen Änderung/unbefugte Benutzung
  - . Schreibschutz bei PSD
  - . Schutz der HSP-Adreßbereiche gegeneinander (jeder Zugriff)
- ÜBERWACHEN VON  
OPERATIONEN UND  
ABLÄUFEN
- : Zeitüberwachung
    - . begrenzt ununterbrochene Benutzung des Zentralprozessors durch ein Programm (CPU-Zeitscheibe)
    - . begrenzt Verweilzeit der PRP in den Laufbereichen
    - . begrenzt durch vom Monitor angestoßene Funktionen
    - . begrenzt den Geräteverkehr
  - : Transferüberwachung
    - . verhindert Dateneingabe in geschützte Bereiche

RETTEN UND  
REGENERIEREN VON  
DATEN

- : Systemwiederanlauf
  - . automat. bei Spannungswiederkehr } (Bedingungen
  - . automat. bei Listenüberlauf des ORG } s. 11)
  - . mit Aufruf auslösbar
  - . manuell auslösbar (Bedienung an die virtuelle Konsole im STOP-Zustand)
- : Neustart, sofern generiert (alternativ zum Wiederaufsetzen)
  - . automatisch bei Spannungswiederkehr (bei HSP-Pufferung)
  - . automatisch bei Listenüberlauf
  - . manuell auslösbar (Bedienung an die virtuelle Konsole im STOP-Zustand)
- : Wiederaufsetzen (bei HSP-Pufferung) alternativ zum Neustart, automatisch bei Spannungswiederkehr am ZE-Rahmen, sofern generiert
- : Abbild des Anwendersystems für Wiederanlauf anlegen (Aufruf)
- : Retten von HSP-Bereichen vor Wiederanlauf zur Fehlerüberprüfung (post-mortem-review) und Datenaktualisierung (Systemdaten)
- : Puffern der E/A-Aufrufe bei Spannungsausfall am Erweiterungsrahmen

SERVICE

- HW/SW-Service-Unterstützung über allgemein verwendbare Schnittstellen wie
- Prüfen und Verwalten von Geräten, die vom ORG über Service-Treiber betrieben werden
  - Prüf-ORG für den Ablauf von HW-Testprogrammen
  - Erfassen von Betriebsdaten eines Gerätes (Inline-Statistik)
  - Testpunkte/Diagnosetestpunkte
  - Diagnosedaten nach Systemausfall
  - Teleservice
  - Paßwort für virtuelle Konsole, Teleservice
  - Fehlerfrüherkennung durch Anlauf- und Selbsttest

1.2.7 Hilfs- und Zusatzfunktionen

- SYSTEM-  
INBETRIEBNAHME
- umfaßt die Überführung der Software von der Lieferform in den Ablaufzustand
- . Betriebssystem (Erstlieferung) BS-M oder AMBOSS 4 nach Kundenwünschen bestellbar, Anlauf per "Knopfdruck"
  - . System-Generieren komfortables Erstellen und Modifizieren eines Betriebssystems (BS-M, AMBOSS 4) mit AMGENM,
    - : Generieren von
      - . Standardkonfigurationen (voll automatisch)
      - . Sonderkonfigurationen (dialoggesteuert)
    - : Integration von bereits gebundenen Modulen in das Betriebssystem
    - : Übernahme von Nachlieferungen und Korrekturen
  - . Grundbetriebssystem HSP-residentes Organisationsprogramm fester Funktionsausstattung. Ermöglicht Ablauf der Inbetriebnahme-Dienstprogramme zur Vorbereitung und Durchführung der ORG-Generierung.
  - . Peripherespeichervorbereitung Formatieren und Einrichten der Datenträger, Anlegen der Dateibuchführung mit Dienstprogramm INITM.
  - . ORG-Generieren Erstellen des ablauffähigen ORG aus dem ORG-Masterstapel mittels unter dem Grundbetriebssystem ablaufenden ORG-Generator über Generierparameter für
    - . Systemspeicherangaben
    - . Anlagenausstattung
    - . Bausteinwünsche
  - . Laden Laden und Löschen von Grundsprache-Anwender-Objekten (HSP, PSD) per Aufruf und Kommando, Laden segmentierter Programme und einseitiges Binden, Erstellen von Arbeitsformen und Laden aus Arbeitsformen
  - . Einrichten globaler Datenbereiche Dynamische Datenfeldverwaltung für temporären Hauptspeicherplatz
  - . Abbild anlegen Zur Sicherung der Software für den Wiederanlauf nach Systemausfall, Abbild enthält
    - . ablauffähiges ORG mit Systemdateien und aktueller Buchführung
    - . Anwenderobjekte
  - . Systemkorrektur : Online-Einbringen von Korrekturen in das ORG-Abbild, Aktivierung durch Wiederanlauf
- SYSTEM-  
INFORMATIONEN
- Über Kommandos oder Aufrufe abrufbar
- . ORG-Daten per Aufruf in Anwenderpuffer übergebbar sind
    - . Systemkennwerte, Listenadressen, Zeitparameter

- 
- . Systemdaten per Kommando protokollierbar sind
    - : Art, Länge, Lage, Priorität, Zustand von
      - . Programmen, Common Data, Common Code, Laufbereichen Paketen
    - : Informationen über
      - . Startlisten, Koordinierungszähler
  
  - . Auslastungs- statistik liefert per Aufruf oder Kommando
    - . ORG-Laufzeiten
    - . Anwenderlaufzeiten
    - . Untätigkeitszeiten
    - . Transferdaten für PSD
    - . Poolbelegung
    - . PAGE- und Cache-Statistikdaten
  
  - ZEITFUNKTIONEN ORG bildet Datum und Uhrzeit mittels Hardware-Zeitgeber, daneben
    - . Relativzeitählung
    - . Zeitmodifikationen für Aufrufe
  
  - SYSTEM-NAHTSTELLEN : Steuerung des Ablaufs von Testhilfen und Testobjekten
    - : Funktionen für den Monitorbetrieb
  
  - SYSTEMS IMULATIONEN . Simulation ZEO3-Befehle für ZEO1
    - . z.T. Festpunktarithmetik lang (Typ D)
    - . Gleitpunktarithmetik kurz (Typ G)
    - . Gleitpunktarithmetik lang (Typ K) } , generierbar
  
  - KONVERTIERUNGEN . Konvertierungsroutinen, immer vorhanden
    - . Code-Konvertierungen (ASCII <-> EBCDIC.S), generierbar
    - . Hilfsroutinen, immer vorhanden

### 1.2.8 Aufrufe und ORG-Kommandos

- AUFRUFSTRUKTUR : ORG-Aufrufe umfassen bis zu 3 Elemente (als Makros verfügbar) für
    - . Anstoß (Befehl)
    - . Parameterblock (Datenfeld)
    - . Gerätedateiblock (Datenfeld)  - : Makroaufrufe außerhalb der ORG-Funktionen für
    - . Konvertierungen
    - . Hilfsfunktionen  - : Makrodefinitionen stehen als Makrobibliothek zur Verfügung
- 
- AUFRUFFUNKTIONEN
  - . Programmablauf- steuerung Programm starten\*)/anhalten\*)/fortsetzen\*)/beenden\*)  
Warten auf Operationsende/auf Fortsetzung  
Startsperre setzen/löschen  
Platzwechselsperre setzen/löschen  
Platzwechsel/Programmwechsel anfordern

Laufbereichsstruktur ändern  
PRP-Residenz und -Fixierung  
Zeitscheibenbearbeitung  
Koordinierungszähler einrichten/betätigen\*)/löschen  
Programmzustand "ausgetestet" einstellen/aufheben

\*) zeitmodifizierbar

- . Datenübertragung Datenfeld lesen/schreiben
- . Geräteverkehr Standard-Eingabe/Ausgabe  
Peripheralspeichertransfer  
Magnetband-Eingabe/Ausgabe/-Steueroperationen  
Magnetbandkassetten-Operationen
  - . Rückspulen auf Bandanfangsmarke
  - . Umspulen auf BandanfangsmarkeDatenübertragungssteuerung
  - . Eingabe/Ausgabe
  - . Parametrierung
  - . Geräteanzeigen erwarten
  - . Urladeanstoß
  - . Sperren/FreigebenGerät offline/online schalten  
Schreibschutz für PSD ein-/ausschalten  
Cache-Betrieb steuern  
Transferwarteschlange löschen  
Gerät zuteilen  
Gerät zuordnen (Anwendertreiber, Servicereaktionsprogramm)  
Alarm anmelden/abmelden  
alarmgesteuerte Koordinierung an-/abmelden  
Gerät basisparametrieren/rücksetzen  
Geräte parametrieren  
Datenträger einrichten
- . Dateiverkehr Datei einrichten/eröffnen/schließen/löschen  
Dateizeiger einstellen/lesen  
Dateilänge ändern  
Dateietikett lesen/schreiben  
Dateibuchführung lesen
- . Zeitbearbeitung Datum und Uhrzeit lesen/einstellen  
Zeitmodifizierte Aufrufe zurücknehmen  
Relativzeitzähler einstellen/ausgeben
- . Bedienen Bedienbereitschaft  
Bedienungsübergabe  
Programm-Geräte-Zuordnung
- . Wiederanlauf Wiederanlauf auslösen  
Rettdatetei freigeben
- . Laden und Sichern Objekt laden/löschen  
Paketobjekte schützen  
Abbild anlegen

---

|                              |  |
|------------------------------|--|
| . Hilfstätigkeiten           | Feldbereich definieren/löschen<br>Feld anfordern/freigeben<br>Programm- oder Testparameter übergeben<br>Objektadresse ermitteln<br>Common-Data-Adresse ermitteln<br>ASCII-EBCDIC.S-Umcodierung<br>Datenfelder rücksetzen<br>Gerätedateiblöcke anlegen<br>Aufrufe für Testprogramme<br>Aufrufe für Monitorbetrieb<br>Diagnose, Statistik und Test für Geräte<br>Firmware laden<br>Geräteprozeßblock laden<br>Auslastungsstatistik abrufen/löschen<br>ORG-Daten ausgeben   |
| KOMMANDOS                    | Mit einer Kennung sind diejenigen Kommandos des Standardbedienprogramms (SBP) versehen, die auch im Mini-SBP in vollem (*) oder eingeschränktem (e) Funktionsumfang zur Verfügung stehen.  |
| Programm- und Zeitfunktionen | (e) Programm starten/beenden<br>Startsperre setzen/löschen<br>(e) Datum und Uhrzeit einstellen/lesen<br>Zeitmodifizierte Aufrufe zurücknehmen<br>(e) CPU-Zeitscheibe ein-/ausschalten, verändern<br>Anwendertreiber-Geräte-Zuordnung eintragen/löschen<br>Laufbereichsstruktur temporär ändern<br>Programm auf Laufbereich fixieren<br>Name-Nummern-Zuordnung eintragen/löschen/ausgeben<br>Programm-Geräte-Zuordnung eintragen(*)/löschen(e)<br>Datensichtstation-Programm-Zuordnung ein-/ausschalten<br>Programmmzustand "ausgetestet" einstellen/aufheben   |
| Laden, Sichern, Geräte       | Objekte laden(e), laden und starten, löschen(e)<br>Gerät klar/unklar erklären<br>(* ) Gerät offline/online schalten<br>Terminalbetrieb ein/ausschalten<br>Meldeverhalten von Gerätefehlern<br>E/A-Zeitüberwachung steuern<br>Steuerkartenbedienung einschalten<br>Paketenschutz ein-/austragen<br>Voreinstellung für Arbeitsformgerät vornehmen/zurücknehmen<br>Namen der Arbeitsform modifizieren<br>Laden aus Arbeitsform ein-/ausschalten<br>Arbeitsform für lineares Programm anlegen<br>V-Teil-Verlängerung auf Laufbereichslänge vereinbaren<br>V-Teil-Verlängerung zurücknehmen<br>(* ) Laufwerk offline/online schalten<br>Schreibschutz für PSD-Laufwerk ein-/ausschalten<br>Cachebetrieb steuern |

---

|                             |  |
|-----------------------------|--|
| Bedienen,<br>Protokollieren | Protokolliergerät einstellen, Protokollierung steuern<br>Bedienzeichenersatz laden/löschen/protokollieren<br>(* Meldung ausgeben<br>Batchbetrieb einschalten/beenden<br>Batchbetrieb nach Wiederanlauf vormerken/zurücknehmen<br>Kommentar im Batchbetrieb<br>Batchbetrieb unterbrechen/fortsetzen<br>Steuerkarten-Protokollierung unterdrücken/fortsetzen<br>Logbuchfunktionen  |
| Systemdatenausgabe          | Informationen ausgeben über Systemobjekte,<br>alle/hauptspeicherresidente(e) Programme, Common Data,<br>Common Code, Common Data im Commonbereich, Laufbereiche,<br>Pakete, Startlisten, Koordinierungszähler.<br>Auslastungsstatistik abrufen/abschalten  |
| Wiederanlauf                | Monitorbetreuung<br>Abbild anlegen mit/ohne Wiederanlauf<br>Wiederanlauf auslösen<br>Abbild in Urladeformat umwandeln<br>Startliste aufbauen/schließen<br>Startliste aktivieren/nicht aktivieren<br>Daten in Parametrierfeld eintragen<br>Parametrierfeld protokollieren<br>in Hauptspeicherbetrieb umschalten<br>ausschließlich Mini-SBP: in Peripheriespeicherbetrieb<br>umschalten<br>Koordinierungszähler einrichten/löschen<br>Dateiparameter angeben<br>Datei einrichten/schließen/löschen |
| Teleservice                 | (* Fernbedienung von Programmen über Teleservice freigeben/<br>sperrern (ggf. auch im Abbild vermerken)<br>(* Paßwort ändern (ggf. ins Abbild übernehmen)<br>(* Gerätenamen der Datensichtstationen ausgeben<br>(* Mithören ermöglichen  |
| Auskunft,<br>Protokoll      | aktuelle Vorbesetzungen<br>Kommandoauskunft<br>Geräteauskunft über Bediengerät<br>(* aktuelles Standardbedienprogramm beenden<br>Protokollausgabe steuern/abbrechen  |

### 1.3 Identifikationskennungen

In den Systemen gibt es Kennungen zur Unterscheidung der Zentraleinheiten, des Organisationsprogramms und der Systemdateien. Die Kennungen lassen sich durch Bedienen über virtuelle Konsole, per Befehl oder mit Aufruf ermitteln.

- Maschinen-  
kennung Die Maschinenkennung dient den hardwarenahen Programmen zur Unterscheidung der Zentraleinheiten. Die Kennung steht im Wort MAKENN mit der Adresse 1024 (dezimal 400) des Hardware-Verständigungsbereichs, ist also stets reell zu adressieren. Wie die Kennung selbst lautet, kann der Anwender der ZE-Beschreibung entnehmen (s. /25/, /26/).
- ORG-Kennung Die ORG-Kennung bietet eine Aussage über das im Hauptspeicher geladene ORG. Sie kann von Anwenderprogrammen mit einem der Aufrufe \$PROPAR oder \$ORGDAT (s. /101/) ermittelt werden. Die Kennung selbst lautet 'H=7000' für das ORG-M.
- System-  
kennzeichen Die bei der ORG-Generierung entstehenden Systemdateien (s. 10) erhalten Namen nach folgendem Bildungsgesetz: YXs... Zur Unterscheidung verschiedener Software-Systeme ist das dritte Zeichen des Namens ("s") vom Anwender bei der ORG-Generierung vorzugeben. Der Anwender kann es später zur Identifikation von Systemdateien mit heranziehen. Für "s" sind die Großbuchstaben A bis Z und die Ziffern 0 bis 9 zugelassen.



---

## 2 Verwaltungsobjekte

Verwaltungsobjekte des ORG sind Ablaufobjekte, Geräte, Dateien, Speicherbereiche und Pakete. Zugeordnete Eintragungen im ORG entstehen beim Generieren und Laden.

### 2.1 Ablaufobjekte, Programme

Ablaufobjekte sind Befehlsfolgen mit zugehörigen Daten, die der Verwaltung des ORG unterliegen. Es sind Programme (s.u.) und bedingt Anwendertreiber (s. 5.6).

Ein Programm ist eine Folge von Befehlen und Daten, die als kleinste Einheit selbständig ablauffähig ist. Es erhält vom ORG Betriebsmittel zugeteilt und kann simultan zu anderen Programmen ablaufen (s. 4).

Programmarten Nach der Art der Abspeicherung und des Ablaufs zu unterscheiden (Ladeparameter):

HRP = hauptspeicherresidente Programme stehen ständig im HSP an den ihnen beim Laden zugeteilten Plätzen, auch wenn sie ruhen.

PRP = peripherspeicherresidente Programme sind im Wartebereich (Datei) abgelegt und werden nur zum Ablauf in HSP-Laufbereiche geholt.

Lange Programme lassen sich bei Speicherplatzknappheit oder Überschreitung des Adressivolumens segmentieren. Sie bestehen dann aus einem Rumpfstück (Steuerteil, Root-Segment) und mehreren Teilstücken (Segmenten). Ein segmentiertes Programm läuft gestaffelt ab. Der Steuerteil veranlaßt nacheinander die Aktivierung der Segmente oder Segmentketten (s. 4.9).

Der Binder erlaubt zwei Arten der Segmentstruktur:

- Normale Rasterung der Segmente. Sie stehen in der Arbeitsformdatei. Eine vom Lader in das Programm eingebaute Segmentwechselroutine übernimmt den Eintransfer der jeweils benötigten Segmente.
- Segmente in Seiten gerastert und ständig im HSP. Das ist für Programme erwünscht, die länger als 65536 Wörter sind, aber als HRP schnell ablaufen sollen. Dabei wird der Zugriff zu Segmenten im fremden Adreßraum vom ORG durch Umladen der Segment-Seitenadressen in der Übersetzungstafel vorgenommen. Programmkurzbezeichnung: SHP.

Programme können in Pakete oder in eigene Bereiche geladen werden. An Laufbereichen stehen Paket-, alternative und selbständige Laufbereiche zur Verfügung.

Die folgende Tabelle zeigt alle Arten von Ladeobjekten, einschließlich der Common-Objekte:

| Objektarten     | in Paketen |             | selbständige Objekte |                        |            | im Common-bereich |
|-----------------|------------|-------------|----------------------|------------------------|------------|-------------------|
|                 |            | Laufbereich |                      | alternat. Laufbereiche | selbständ. |                   |
| linear          | HRP        | PRP         | HRP                  | PRP                    | PRP        | -                 |
| segmentiert     | HRP        | PRP         | HRP                  | PRP                    | PRP        | -                 |
| HSP-segmentiert | -          | -           | SHP                  | -                      | -          | -                 |
| Common-Objekte  | CD         | -           | CC                   | -                      | -          | CD                |

**Aufbau**

Ein Programm besteht aus Kopf und Rumpf. Den Kopf bilden Übersetzungstafel (ÜT) und Parametertafel (PT). Beide können vom Rumpf getrennt abgespeichert sein oder ihm direkt vorausgehen. Jedes Programm hat seine eigene PT. Sie enthält Buchführungsinformation des ORG und das Abbild der Standardregister. Der Lader richtet die PT ein (s. 12). Ein Programm hat seine eigene ÜT oder teilt sie - in Paketen - mit anderen.

Im Programmrumpf stehen Code und Daten. Übersetzungsprogramme teilen beim Übersetzen den Rumpf in einen varianten Teil (Daten) und einen invarianten Teil (Code, Konstanten) auf. Die Aufteilung richtet sich nach der vom Programmierer angegebenen Programmteilkennung ('I' oder 'V'). Benutzung von mehreren Adreßräumen s. 4.7 (TZ2) und /216/ (Adreßraumerweiterung).

**Identifikation** Ein Programm wird durch eine vom Anwender beim Laden anzugebende oder vom Lader vergebene Nummer gekennzeichnet. Sie muß innerhalb des festgelegten Nummernbereichs liegen und darf außerdem nicht schon durch ein geladenes/eingerichtetes Objekt belegt sein.

Ist einem bestimmten Objekt (Programm/CD/CC) auch ein Name zugeordnet (Namen-Nummern-Zuordnung), kann der Anwender in Bedienungen dieses Objekt statt durch die Objekt-Nummer auch durch den Objekt-Namen identifizieren.

**Priorität**

Jedes Programm erhält zusätzlich eine Priorität, nach der Zuteilungen des Zentralprozessors und Warteschlangen-Bearbeitungen erfolgen. Die Priorität entspricht den Ebenenprioritäten des Zentralprozessors, die für die Anwenderprogramme zur Verfügung stehen, und kann demnach Nummern zwischen 5 und 15 (= niedrigste Priorität) annehmen. Die Priorität ist frei wählbar, kann beim Laden des Programms angegeben und beim Start verändert werden. Mehrere Programme können die gleiche Priorität erhalten, ihre Behandlung erfolgt dann in der zeitlichen Reihenfolge der Anforderungen.

Die in den Vorläufersystemen benutzten Softwareprioritäten mit dem Nummernbereich 1 bis 255 können in Aufrufen weiter benutzt werden. Solche Angaben wandelt das ORG gemäß folgender Tabelle um. Zu beachten sind bei Übernahme bisheriger Programmsysteme mögliche Ablaufinkompatibilitäten (s. /2/) und die geringere Stufung.

| Nummern der Software-Priorität                       |             | Priorität          | Priorität                  |
|--|-------------|--------------------|----------------------------|
| 1  | bis p1      | 1 - 23             | 5                          |
| p1   | + 1 bis p2  | 24 - 47            | 6                          |
| p2   | + 1 bis p3  | 48 - 71            | 7                          |
| p3   | + 1 bis p4  | 72 - 95            | 8                          |
| p4   | + 1 bis p5  | 96 - 119           | 9                          |
| p5   | + 1 bis p6  | 120 - 143          | 10                         |
| p6   | + 1 bis p7  | 144 - 167          | 11                         |
| p7   | + 1 bis p8  | 168 - 191          | 12                         |
| p8   | + 1 bis p9  | 192 - 215          | 13                         |
| p9   | + 1 bis p10 | 216 - 239          | 14                         |
| p10  | + 1 bis 255 | 240 - 255          | 15                         |
| Umrechnungsvorgabe durch den Generierparameter /PRIO |             | Standardumrechnung | höchste<br>A<br>niedrigste |

Beim Generieren kann der Anwender eine individuelle Einteilung der Softwarepriorität für die Umrechnung vorgeben.

Verwaltung,  
PNULI

Zur Verwaltung von Programmen dient dem ORG im wesentlichen die Programmnummernliste (PNULI, s. 4.1). Sie führt Programme, Common Data und Common Code (s. 2.4) gemeinsam und enthält je Objekt (PNULI-Element) u.a. Objektart, Objektname, Programmzustand, Priorität und Speicherort. Die Objektzahl entspricht dem Platz in der PNULI und muß deshalb eindeutig einem Objekt beim Laden zugeordnet werden (s. 12).

Unterbrechbarkeit

Ein ablaufendes Objekt ist - sofern nicht durch gesetzte Zustandswechselferriere verhindert - nach jedem Befehl bzw. Befehlsschritt (s. /25/, /26/) durch höherprioritäre Anforderungen unterbrechbar (periphere Anforderungen, ORG-Aufträge, höherprioritäre Programme).

## 2.2 Geräte

Gerätetypen

Es können zwei Hauptgruppen von peripheren Geräten unterschieden werden: Standard- und Prozeß-Peripherie (s. 5). Daneben unterliegen Geräte der Sonderperipherie einer besonderen Behandlung (s. 5.6).

Standard-Peripherie

Zur Standard-Peripherie gehören die Geräte gemäß 1.2.5. Diese Geräte können über sogenannte Standard-Ein-/Ausgabe-Aufrufe und diesen ähnliche E/A-Aufrufe vom Programmierer angesprochen werden (s. 5.2, 5.3).

Bei geeigneten Geräten lassen sich Daten sowohl alphanumerisch (mit Steuerzeichenerkennung) als auch binär ein- oder ausgeben (s. 5.2, 5.3). Der Anwender verkehrt mit Plattenspeichern vorwiegend über die Dateiorganisation (s. 2.3). Standard-E/A-Verkehr mit PSD ist nur bei eingerichteten Datenträgern möglich.

Die Funktionen und Möglichkeiten der Geräte lassen sich durch ladbare Firmware verändern und erweitern; Firmware-Verwaltung s. /323/.

- 
- . Virtuelle Geräte  
Das Spoolsystem SPOOL-M (s. /402/) arbeitet mit virtuellen Geräten, deren Gesamtzahl beim Generieren anzugeben ist: logische Geräte, die weder generiert werden noch installiert sind, auf die aber E/A-Aufrufe Bezug nehmen. Ein PSD dient als Zwischenspeicher für die zu transferierenden Daten. Zeitlich entkoppelt werden diese an einem (realen) Gerät ausgegeben.
  - . Globale Geräte  
Im Rechnerkommunikationssystem können globale Geräte vereinbart sein, deren Gesamtzahl beim Generieren anzugeben ist. Ein globales Gerät, das einem bestimmten Gerät an einem Rechner eines Rechnernetzes zugeordnet ist, kann der Anwender in E/A-Aufrufen benutzen, um von einem anderen Netz-Rechner aus mit dem Gerät zu verkehren.
  - . Prozeß-Peripherie  
Prozeßperipherie kann nur mit geräteabhängigen Aufrufen angesprochen werden. Sie laufen in einigen Fällen (sogenannte "schnelle" Prozeßsignalformer) nicht über das ORG. Es treten Folgen von Ein-Ausgabe-Maschinenbefehlen im Anwenderprogramm auf.  
  
Einige Prozeßgeräte (integrierende Analogeingabe, Alarmeingabe u.a.) geben zeitlich unbestimmte Meldungen (Alarmer). Anwenderprogramme können auf solche Anforderungen warten. Das ORG erkennt eintreffende Alarmer und setzt das zuständige Programm fort (s. 5.5).
- Identifikation Der Anwender verkehrt mit logischen Geräten und identifiziert sie durch logische Gerätenamen. Dieser Name ist unabhängig von der physikalischen Belegung der Anlage. Er enthält eine vorgeschriebene Kennung des Gerätetyps und ein oder zwei Nummern als Unterscheidungsmerkmal für Geräte gleichen Typs.  
  
Das Gerät mit seiner physikalischen E/A-Adresse wird dem ORG beim Generieren mitgeteilt und in der Geräteliste dem logischen Gerätenamen zugeordnet. Der logische Gerätenamen ist zum Generierzeitpunkt beliebig änderbar. Die logischen Geräte entsprechen bei PSD den Teilspeichern.
- Geräte-Verwaltung Periphere Geräte sind vom ORG verwaltete Betriebsmittel, die den Programmen auf Anforderung zugeordnet werden. Die Zuordnung bei mehreren Anforderungen für dasselbe Gerät erfolgt in prioritätsgesteuerter Reihenfolge.  
  
Zu jedem an einer Anlage angeschlossenen peripheren Gerät enthält das ORG einen Geräteblock. Die darin eingetragenen Informationen entstehen aus Anwenderangaben beim Systemgenerieren. Die Summe dieser und der virtuellen und globalen Geräteblöcke bildet die Geräteliste des ORG.  
  
Zum Verkehr der Programme mit Datensichtstation und Drucker ist eine Datenpufferung vorgesehen, die vorzeitige Laufbereichsräumung von PRP und schnellere Programmabläufe ermöglicht (s. 5.4). Im Gegensatz zur Standard- und Prozeßperipherie führt das ORG den E/A-Verkehr mit Sonderperipherie nicht durch (s. 5.6).

---

## 2.3 Dateien, Bibliotheken

- Zusammenfassung** Eine Datei ist eine symbolisch benannte, vom ORG verwaltete Datenmenge auf einem Datenträger. In der Regel werden zusammengehörige Daten vereinbarter Struktur, Datendarstellung und Reihenfolge eine Datei bilden.
- Dateien sind - wie Ablaufobjekte - vom Anwender zu planen, einzurichten und dem ORG mitzuteilen. Mit dem Einrichten sind Speicherplatzzuweisung und Buchführungseintragungen verbunden. Nach dem Einrichten ist der Dateiinhalt undefiniert (s. 6).
- Dateiarten** Das ORG kennt nur peripherspeicherresidente Dateien (PRD) auf PSD. Strukturierte Daten im HSP sind als Common Data oder programmspezifische Datenfelder vorzusehen.
- PRD bleiben über einen Wiederanlauf hinweg ohne besondere Maßnahmen erhalten. Nicht mehr benötigte Dateien lassen sich mit einem Aufruf löschen. Die Buchführung zu PSD-Dateien steht auf dem Datenträger, Teilabbilder davon fallweise auch im HSP. Solange zugehörige Systemkonventionen eingehalten werden, lassen sich Wechselplatten oder Disketten mit Dateien jederzeit austauschen und von den Programmen mit Datei-Aufrufen ansprechen.
- Dateiaufbau** Eine Datei besteht aus Dateietikett/-kennsatz und Daten, die räumlich getrennt auf dem PSD abgespeichert sind. Das Etikett identifiziert die Datei und beschreibt ihre Inhaltsstruktur und Lage.
- Die Daten können in logische Sätze und/oder physikalische Blöcke gegliedert sein. Logische Sätze sind Längenabschnitte innerhalb der Datei, an denen sich der Software-Zugriff orientiert. Physikalische Blöcke sind durch Hardware-Eigenschaften des Ein-Ausgabe-Geräts stets geschlossen zu verarbeitende Datenmengen. Die Daten einer Datei können alphanumerisch oder binär sein.
- Der Anwender benutzt für den Verkehr mit Dateien Standard-Ein/Ausgabe-Aufrufe und kann zwischen direktem und seriellem Zugriff wählen (s. 6).
- Datei-Identifikation** Dateien werden durch die Angabe des Gerätes (logischer Geräte-Name) und des Dateinamens identifiziert.
- Datenteilmengen innerhalb der Datei lassen sich durch Angabe expliziter Adressen ansprechen (Dateizeiger).
- Datei-Verwaltung** Das ORG steuert den Zugriff der Benutzer zu den Dateien ähnlich wie bei Geräten. Zur Buchführung dienen dem ORG Datenträgeretiketten, Speicherbelegungstafeln, ein Dateikatalog (bestehend aus Dateietikettbereich und Dateinamens-tafel) sowie eine Dateinamensliste für die gerade bearbeiteten (eröffneten) Dateien.

Bibliotheken Für die systematische Verwaltung von Datenmengen auf Datenträgern und den Verkehr mit ihnen besteht neben der Dateiorganisation ein Bibliothekssystem. Es ist der Dateiorganisation übergeordnet. Es erlaubt gegenüber Dateien eine zweistufige Namensvergabe, ermöglicht die Führung mehrerer Ausgabestände des gleichen Objektes, verbietet aber Binärdaten. Ein Bibliotheksbenutzer braucht Dateiorganisation und zugeordnete Aufrufe nicht zu kennen.

Das ORG kennt und verwaltet Bibliotheken nicht. Der Lader bietet aber als Teilleistung das Laden aus Bibliotheken (s. 12). Assembler, Compiler und manche Dienstprogramme organisieren ihren Datenzugriff nach den Bibliothekskonventionen, die Libraries höherer Programmiersprachen werden z.B. als Bibliotheken geführt. Ferner stehen als Dienstprogrammfunktionen für das Arbeiten mit Bibliotheken das Ein-Ausgabe-System BIBEAS-M und das Dienstprogramm FILE-M zur Verfügung (/319, 303/).

Bibliotheken enthalten Datensätze aus alphanumerischen Zeichen (Quellsprache, Grundsprache oder Texte), die in Bibliothekselementen gruppiert auf einem Random-Datenträger mit zugeordneter Bibliotheksbuchführung abgelegt sind. Die Bibliotheksstruktur stützt sich auf die Dateiorganisation des jeweiligen ORG ab: Bibliotheks-Buchführung und -Elemente sind Dateien.

Daten in Bibliotheken werden identifiziert durch den log. Gerätenamen, den Bibliothekstyp (Quellsprache, Grundsprache, Texte), den Bibliotheksnamen, den Elementnamen und den Ausgabestand.

DVS Das Datenverwaltungssystem DVS-M (s. /401/) verwaltet Dateien nach Suchbegriffen (Schlüsseln). DVS-M benötigt im ORG eine Nahtstelle, die der Generator auf Anwenderwunsch installiert.

#### 2.4 Gemeinsame Daten und Felder

Common Data Common Data sind HSP-Bereiche für Daten. Sie werden ähnlich  
CD wie Programme geladen und verwaltet, benötigen keine PT und dienen u.a. dem schnellen und einfachen Datenaustausch zwischen Programmen. Virtuelle Lage und Länge gibt der Benutzer beim Laden an. Auch Code-Teile - z.B. Unterprogramme - sind in Common Data möglich. Zugriffe zu CD s. 4.8.1.

- Paket-CD stehen in beliebiger Mischung mit den HRP in Paketen vor den Laufbereichen. Paket-CD sind über die Paket-ÜT zugänglich.
- Bereichs-CD stehen im dafür vorgesehenen Commonbereich (s.u.).

Commonbereich Je System ist ein Commonbereich zur Aufnahme von CD möglich.  
CB Die gewünschte Gesamtlänge und die Länge eines davon zu schützenden hinteren Teils sind bei der Generierung anzugeben.

Die Lage im Speicher legt der Generator im Anschluß an die Pakete fest. Die Platzaufteilung im CB entscheidet der Anwender beim Laden der CD oder der Lader selbsttätig.

Für den CB besteht im ORG eine Übersetzungstafel. Diese ÜT wird für den Zugriff eines Anwenderprogramms zu einem CD in die ÜT des rufenden Programms/Pakets übertragen. Dann gelten die Einträge am Anfang der ÜT für die vom Programm bzw. den Paket-Objekten belegten Seiten und am Ende der ÜT für die Seiten des CB. Bei Platzmangel werden nur die letzten CB-ÜT-Seiten übernommen. Demgemäß ist die Platzvergabe im CB mit den rufenden Programmen abzustimmen. Als Vorkehrung zum Ladebinden füllt der Generator bereits die Paket-ÜT (s.u.) mit dem CB-ÜT-Seiten auf.

Common Code  
CC

Codeteile mit variantem Teil, der mehrfach benutzbar sein soll, können als selbständige Common Codes (CC bzw. SCC) geladen werden. Der Lader lädt den invarianten Teil Hauptspeicherresident und hinterlegt den varianten Teil in einer - dem CC zugeordneten - Hilfsdatei (s. 12.3). Es können mehrere CC in einem System existieren.

CC können als gemeinsame Bereiche für mehrere Programme oder als Erweiterung des Programms (Code oder Daten) benutzt werden (Adreßraumerweiterung s. /216/).

CC können in vielfältiger Weise zu anderen Objekten ladegebunden bzw. gekoppelt werden (mit anderen CC, selbständigen HRP/PRP und dem Commonbereich). Ein CC (reentrant) kann beim Laden zu Paketen gekoppelt werden. Beim Koppeln eines selbständigen Programms zu einem CC wird der in der CC-Datei hinterlegte variante Teil des CC in das koppelnde Objekt kopiert (Vergrößerung der Programmlängen beachten). Der I-Teil wird ähnlich wie CD zum Programm ladegebunden. Zugriffe auf CC s. 4.7 und 12.

Feldeinheiten

Feldeinheiten sind vom ORG (wie CD) verwaltete gemeinsame Speicherplätze im HSP, auf die der Anwender von allen Programmen aus zugreifen kann. Im Gegensatz zu Common Data wird in diesen Speicherplatz nicht geladen. Der Anwender richtet Feldeinheiten beim ORG-Generieren ein und strukturiert sie (s. /400/).

## 2.5 Pakete

Zusammenfassung

In Paketen lassen sich Programme zu Systemen zusammenfassen, zwecks Übernahme von anderen Systemen und um den HSP-Verbrauch zu verringern. Max. 15 Pakete können entweder als Programmpakete oder als Feldpakete generiert werden; letztere nehmen Feldeinheiten auf und werden hier nicht weiter betrachtet. Aufbau, Speicherort, Platzvergabe s. Bild 1/2 und 1.2.2. Pakete können Ablaufobjekte (HRP, PRP), CD und Feldeinheiten enthalten.

Objekte im Paket sind gegeneinander nicht geschützt. Der Anwender kann aber (außer bei der ZE01) mit einem Aufruf bestimmte Seiten als (schreib)geschützt markieren.

- 
- Adressierung, Ein Paket hat für alle seine Objekte nur eine gemeinsame ÜT.  
ÜT Die max. 15 PT der Paketobjekte stehen hintereinander im Anschluß an die ÜT in der 1. Seite. Diese Seite ist schreibgeschützt (nicht bei der ZEOL). Somit sind zusammen 15 HRP und Laufbereiche möglich. CD in Paketen sind durch diese Begrenzung nicht betroffen. Das ÜT-Ende enthält entweder CB-ÜT-Seiten (Generator) oder die ÜT-Seiten eines CC (CC laden).
- Längen Die Paketlängen (max. 65536 Wörter) sind ganze Vielfache von Seiten. Paket-Laufbereiche sind auf Seiten-Grenzen gerastert.
- Buchführung Das ORG entnimmt Länge und Lage der Pakete aus Generierangaben und registriert die Platzaufteilung im Pakettrumpf gemäß den Anwenderladeparametern.

## 2.6 Laufbereiche, Wartebereich

Laufbereiche (LB) sind beim Generieren vom Anwender einzurichtende HSP-Bereiche, in denen verschiedene PRP abwechselnd ablaufen (s. auch 1.2.2).

- Laufbereichs- - Selbständige LB am HSP-Ende  
arten - Paket-LB am Ende von Paketen  
- Alternative LB vor den selbständigen LB (diese Laufbereiche vergibt das ORG optimierend)  
- Der Standard-Laufbereich ist ein Alternativ-LB oder, wenn solche nicht generiert, der Laufbereich mit der Nummer 1. Der Standard-LB ist Vorbesetzung, wenn beim Laden kein Laufbereich explizit genannt wird, und dient dem Ablauf von Standardbedienprogramm, Lader und Monitor (weshalb er länger als 20 Seiten sein muß).
- Laufbereichs- Insgesamt maximal 255 LB sind möglich; auf selbständige,  
anzahl alternative und Paketlaufbereiche beliebig aufteilbar (Generierangabe).
- Laufbereichs- Laufbereiche werden durch die Laufbereichsnummer identifi-  
identifikation ziert. Sie wird bei der Generierung ab Nr. 1 lückenlos aufsteigend vergeben. Beim Laden wählt der Anwender für sein PRP einen selbständigen oder Paket-LB aus oder gibt vor, daß es in einem Alternativ-LB laufen soll. Alternativ-LB vergibt das ORG platzoptimal; bei jedem Platzwechsel/Eintransfer neu.
- Laufbereichs- Die Längen gibt der Anwender beim Generieren an. Sie betra-  
länge gen stets ganze Vielfache von Seiten, max. 64, in Paketen max. 63. Die Längen der Alternativlaufbereiche können im Betrieb verändert werden (s. 4.6).
- Wartebereich Der Wartebereich ist die Datei auf PSD, die alle PRP aufnimmt. Vom Wartebereich werden die PRP beim Start eintransferiert und bei einer Unterbrechung wieder zurückgespeichert. Einzelheiten zum Platzwechselmechanismus s. 4.3, Festlegung des Wartebereichs durch Generierangaben; die Wartebereichs-Verwaltung übernimmt der Lader.



### 3 Nahtstellen des ORG

Das ORG wird tätig, sobald Anforderungen von Geräten eintreffen oder ORG-Anstöße oder Programmunterbrechungen auftreten. ORG-Tätigkeiten für die Programme werden stets durch einen Informationsaustausch eingeleitet, der über die "Nahtstellen" des ORG erfolgt (s. Bild 3/1).

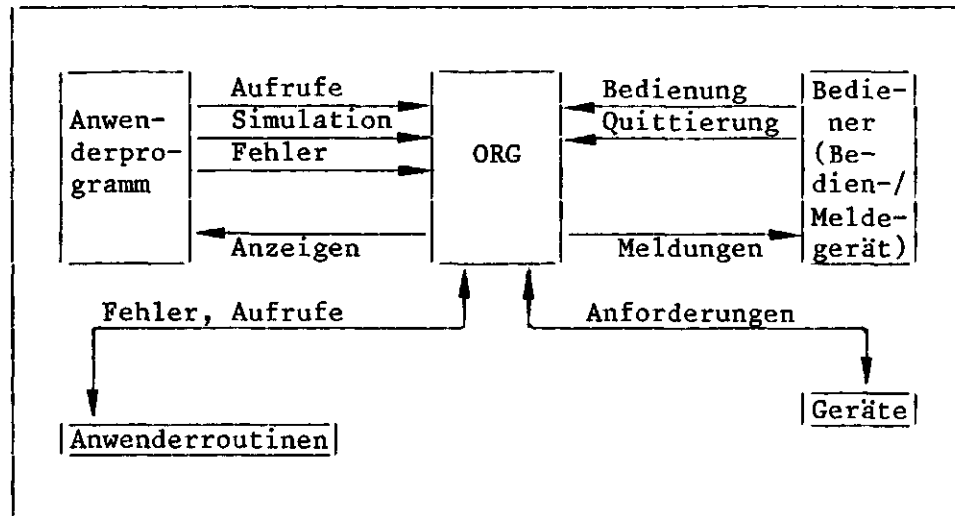


Bild 3/1: Nahtstellen des ORG

Aufrufe bilden die Nahtstelle zu den ORG-Funktionen einschließlich den Ein- und Ausgaben (s. 4, 5 und 6) und zum Laden (s. 12). Nicht interpretierbare Befehle führen zu den Simulationsroutinen (s. 7) und Anwenderfehler zur Fehlerbehandlung (s. 13).

Über Bedieneingaben an Bediengeräten kann der Anwender mit dem ORG und den bedienbaren Programmen verkehren. Der Dialog kann entweder über das Standardbedienprogramm mit Hilfe von Kommandos (s. 9) oder über ein Bediensystem mit Prozedur- und Menütechnik erfolgen (s. /500/).

#### 3.1 Aufrufe

Zusammenfassung

Anwenderprogramme verkehren über Aufrufe mit dem ORG. Diese ORG-Aufrufe sind Anforderungen des Anwenders an das ORG zur Ausführung bestimmter Tätigkeiten. Das ORG gibt den Anwenderprogrammen und ggf. auch dem Bediener Mitteilungen über die Ausführung.

---

|                           |  |
|---------------------------|--|
| Aufruf-<br>struktur       | <p>ORG-Aufrufe setzen sich aus mehreren Blöcken zusammen, die mit Makroaufrufen abgesetzt werden.</p> <ul style="list-style-type: none"><li>- ORG-Anstoß: Dies ist ein spezieller Befehl ("Rufen Primärzustand, RPZ), der als Programmlaufbesonderheit wirkt und den Übergang vom Programm ins ORG auslöst.</li><li>- Parameterblock: Ein Parameterblock beschreibt die gewünschte ORG-Funktion und enthält die nötigen Parameter und die Anzeigenzelle.</li><li>- Geräte-Datei-Block (GEDA-Block): Ein GEDA-Block kennzeichnet das Gerät und die Datei, auf die sich ein Ein-Ausgabe-Aufruf (Parameterblock) bezieht. Insbesondere für Dateien enthält der GEDA-Block auch Buchführungs- und Rückgabeinformationen.</li></ul> <p>Diese Aufteilung der Aufrufinformationen erlaubt die Trennung von Code und Daten und die Trennung von Operation und Gerät (dynamische Gerätezuordnung, Anlagenunabhängigkeit).</p> |
| Rückmeldungen<br>des ORG  | <p>Ist der Aufruf bearbeitet, wird das Programm fortgesetzt. Vom ORG an den Anwender übergebene Informationen sind Anzeigen, Meldungen, Zustandskennbits, Adressen, Fortsetzkennungen, Objektparameter und Wiederanlaufinformationen.</p>  |
| . Aufruf-<br>anzeigen     | <p>Anzeigen sind Bitkennungen, die stets im ersten Wort eines Aufrufparameterblocks hinterlegt werden. Als Sammelanzeige dient Bit <math>\emptyset</math>. Ist es nicht gesetzt, so wurde die gewünschte Funktion ordnungsgemäß ausgeführt. Andernfalls liegen Fehler oder Besonderheiten vor.</p>   |
| . Peripherie-<br>Anzeigen | <p>Bei E/A-Aufrufen treten zusätzlich Peripherie-Anzeigen auf, die das ORG auf Wunsch im GEDA-Block zur Verfügung stellt.</p>  |
| . Meldungen               | <p>Meldungen sind ORG-Mitteilungen an den Benutzer über Textausgabegeräte. Sie können allein oder in Verbindung mit Anzeigen auftreten.</p>  |
| . Objekt-<br>parameter    | <p>Das ORG stellt Aufrufe zur Verfügung, mit denen der Benutzer den Zustand bestimmter Objekte (z.B. eines Programms) erfragen kann (im Parameterblock übergeben). Informationen über Dateien trägt das ORG in den GEDA-Block ein.</p>   |
| Logbuch<br>führen         | <p>Das ORG gestattet es,</p> <ul style="list-style-type: none"><li>- Fehlermeldungen,</li><li>- ORG-Aufrufe (Parameterblock, Programm, Gerät)</li><li>- Ein/Ausgaben an Sichtstationen und Druckern (Aufrufe und Daten)</li></ul> <p>entweder auf einem Logbuchgerät (Datensichtstation, Drucker) laufend auszugeben oder auf der Logbuchdatei zwischenspeichern, um sie später ausgeben zu können. Die Logbuchfunktion ist (außer im HSP-ORG) immer vorhanden, kann aber weggewünscht werden; die Logbuchdatei definiert der Anwender beim Generieren. Zum Steuern und Ausgestalten der Logbuchführung stehen Kommandos zur Verfügung (/SYSLOG..., s. /103/).</p>   |

---

### 3.2 Testnahtstelle

Mit der Testnahtstelle des ORG kann der Ablauf eines Programms (Testobjekt) durch ein anderes Programm (Testhilfe) gesteuert und überwacht (getestet) werden. Folgende Aufrufe stehen zur Verfügung: Testhilfe rufen, Testunterbrechung anmelden und Anmeldung zurücknehmen, Testobjekt fortsetzen. Der ORG-Baustein wurde vornehmlich für die Test-Dienstprogramme geschaffen.

Es können beliebig viele Testhilfen angemeldet werden. Eine Testhilfe kann mehrere Testobjekte gleichzeitig überwachen. Ein Testobjekt kann nur von einer Testhilfe überwacht werden.

### 3.3 Monitornahtstelle

Die Monitornahtstelle stellt Aufrufe zur Verfügung, damit Steuerprogramme (= Monitore) andere Objekte (= Anwenderprogramme) steuern und überwachen können: Objekte ein/austragen, Ereignisse übernehmen, Weckzeit einstellen (s. 21.9 in /101/).

Ein Programm kann zu einer Zeit nur von einem Monitor betreut werden. Mehrere Monitore können je mehrere Programme betreuen.

Im Betriebssystem BS-M kann der Anwender den mitgelieferten System-Monitor MONIM (Programmnummer 4) als Steuerprogramm für den automatischen und seriellen Ablauf von Programmen im Rahmen der Stapelverarbeitung benutzen (s. /306/). MONIM kann mehrmals geladen werden (unter unterschiedlichen Programmnummern). Das Standardbedienprogramm ist als Monitor eingetragen, sobald der Anwender ein geeignetes /MCTRL-Kommando abgegeben hat (s. 2.4 in /103/).

Im Betriebssystem AMBOSS 4 ist ein Monitor integriert; weitere zwei können installiert und betrieben werden (darunter MONIM).

---

### 3.4 Anwendernahtstellen

Bestimmte Anforderungen an Betriebssysteme können nicht universell gelöst werden. Deshalb stellt das ORG Nahtstellen zur Verfügung, zu denen der Anwender spezielle Routinen erstellen und einbringen kann. Einsätze erfordern direkte Projektberatung.

Aufrufbear-  
beitungs-  
nahtstelle

Die Nahtstellenroutine wird bei allen Anwender-Aufrufen verständigt. Die Anwender-Routine kann entscheiden, ob sie den Aufruf ohne Änderungen, mit Änderungen oder überhaupt nicht vom ORG bearbeiten lassen will.

Nahtstelle  
für Fehler-  
meldungen

Die Fehlermeldenahntstellenroutine wird angesprochen, wenn ein meldewürdiges Ereignis auftritt und zwar bevor eine Fehlermeldung ausgegeben wird. Die Anwender-Routine ermöglicht die Änderung von Meldungsausgaben.

Nahtstelle  
für System-  
fehler

Diese Nahtstelle für Systemzusammenbrüche wird bei Listenüberlauf und Segmenttransferfehler verständigt und ermöglicht Wiederanläufe von Ersatzsystemen und Rechnerumschaltung.

#### 4 Programmorganisation

Die Programmorganisation steuert und koordiniert den Ablauf der Programme. Dazu gehören Zuteilung des Zentralprozessors an die Programme, Ausführung der Programmzustands- und Platzwechsel, Bearbeitung der Aufrufe zur Programmkoordinierung, Verwaltung der Programmbuchführung sowie Daten- und Parameterübergabe.

Die Programmorganisation bearbeitet Programme (HSP-residente, HRP und PSD-residente, PRP) und verwaltet auch Common Code (CC) und Common Data (CD). S. auch 2.

##### 4.1 Buchführung

Zur Buchführung über Programme, CD und CC, aber auch Pakete und Laufbereiche dienen dem ORG folgende Listen:

**Programmnummernliste** Sie enthält für Programme, Common Data und Common Code die Objektnamen, Zustände, Adressen u.a. und ist lückenlos nach aufsteigenden Objektnummern sortiert.

Am Ende der Liste sind Elemente angehängt, in denen das ORG seine nichtreellen Adreßräume führt (s. auch 3.4.2 in /103/). Damit sind sie wie Programme ansprechbar (\$TEST, \$NAMNUM, \$RUF OBJ, \$LIES., s. 21.2 in /101/).

Objektname: ORGDA2 für den dritten Datenadreßraum  
              ORGCA2 für den dritten Codeadreßraum  
              ORGDA1 für den zweiten Datenadreßraum  
              ORGCAL für den zweiten Codeadreßraum  
              ORGCOD für den ersten Codeadreßraum

(ORGDAL, ORGDA2, ORGCAL bzw. ORGCA2 entfallen, wenn keine geeignete Adreßraumerweiterung generiert ist.)

**Paketliste** Sie dient der Buchführung über die Paketbereiche.

**Programmparametertafeln (PT)** Programm-PT sind 64 Wörter lang. Der Lader richtet sie ein und stellt ihren Anfangszustand her. Inhalt s. 40.1 in /101/. Die Hardware lädt und rettet Registerinhalte innerhalb der PT bei Zustandswechsel. Die PT steht am Anfang des Programms bzw. des Pakets hinter der ÜT.

**Übersetzungstafeln (ÜT)** Alle selbständigen Programme, Pakete, Laufbereiche, CC und der Commonbereich haben je eine ÜT für die Adressierung (s. /25,26/). Sie steht bei Programmen, CC und Paketen am Anfang, vor der oder den PT. Die CB-ÜT befindet sich im ORG-VB. ÜT sind 64 Wörter lang.

**Laufbereichsliste** Sie führt die Laufbereiche und ihre Zustände und Belegung.

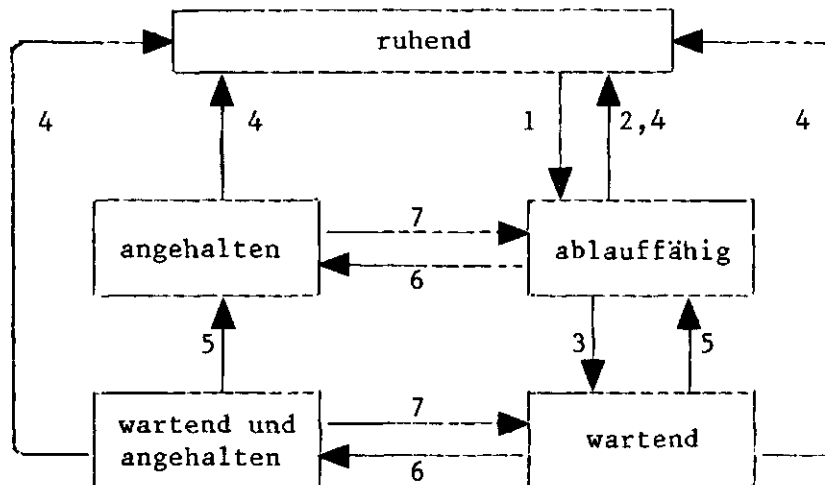
**Warteschlangen** Zur Buchführung über die Programme, ihre Zustände und zur Ablauf-, Laufbereichs- und Platzkoordinierung benutzt das ORG verschiedene Warteschlangen.

#### 4.2 Programmzustände und Zustandswechsel

**Ablaufzustände** Zur Steuerung des Programmablaufs unterscheidet die Programmorganisation für jedes Programm zwischen folgenden Zuständen:

- . **ruhend** Das Programm ist im System vorhanden; es ist geladen, jedoch nicht gestartet bzw. bereits beendet.
- . **ablauffähig** Das Programm ist gestartet. Es wartet auf die Zuteilung des Zentralprozessors oder benutzt ihn bereits. Ablauffähige Programme stehen in der Prioritätswarteschlange nach Priorität, bei gleicher Priorität nach zeitlichem Eingang geordnet.
- . **wartend** Das Programm ist gestartet, aber unterbrochen worden und wartet auf ein Ereignis (Betriebsmittel oder Operationsende). Das wartende Programm wird fortgesetzt (ablauffähig gemacht), wenn das erwartete Ereignis eintritt (Betriebsmittel verfügbar, Operation abgeschlossen, Fortsetzaufruf usw.).
- . **angehalten** Das Programm (zwingend ein HRP) ist gestartet, aber unterbrochen, weil ihm durch ein anderes Programm per Aufruf der Zentralprozessor entzogen wurde. Das kann an einer beliebigen Stelle des Programms geschehen. Der Weiterlauf muß durch ein anderes Programm per Aufruf freigegeben werden. Ein HRP kann gleichzeitig wartend und angehalten sein.

**Zustandswechsel** Die Übergänge zwischen den verschiedenen Programmzuständen organisiert das ORG. Die Ursachen zeigt das folgende Diagramm:



**Ursachen des Zustandswechsels:**

- 1: Start des Programms
- 2: Ende des Programms durch sich selbst
- 3: Warten auf Aufrufende / Fortsetzung
- 4: Beenden des Programms durch ein anderes Programm
- 5: Ein-Ausgabe-Operation abgeschlossen, Aufrufendebearbeitung, Fortsetzung des wartenden Programms
- 6: Anhalten des Programms durch ein anderes Programm
- 7: Fortsetzen des angehaltenen Programms

aktuelles  
Programm,  
Programm-  
wechsel

Dem jeweils höchstprioreren ablauffähigen Programm ist der Zentralprozessor zugeteilt. Es wird auch aktuelles Programm genannt. Die Zuteilung des Zentralprozessors erfolgt hardwaregesteuert. Ändert sich der Zustand des aktuellen Programms (z.B. von "ablauffähig" nach "wartend"), so erhält das nächste Programm der Prioritätswarteschlange den Zentralprozessor zugeteilt. Wird ein höherprioreres Programm ablauffähig, so wird es auch aktuell auf Kosten des bisherigen. Diese Wechsel von aktuellen Programmen werden Programmwechsel genannt.

Programmwechseln gehen stets Wechsel des Programmzustandes voraus. In Verbindung mit dem Programm- und Zustandswechsel tritt für PRP in den Laufbereichen häufig auch ein Platzwechsel auf (s. 4.3).

Programmwechsel zugunsten gleichpriorer Programme ohne auslösende Zustandswechsel s. 4.5 (Funktion "Programmwechsel anfordern").

Zustands-  
modifikation

Speziell für PRP registriert das ORG weitere "Zustände". Die wichtigsten sind:

- Präsenz des PRP (im HSP, im Wartebereich, gerade beim Ein/Austransferieren)
- Sperrung. Ein Programm darf nicht geräumt/verdrängt werden. Dafür sorgen interne Sperrzähler, die das ORG aber auch Anwender betätigen können.
- Laufbereich angefordert
- Austransfer angefordert
- Räumgewicht des Laufbereichs für den Platzwechsel.

Räumgewicht

| Räumgewicht | Das PRP im Laufbereich ....          |
|-------------|--------------------------------------|
| 0           | ist gesperrt (Sperrzähler)           |
| 5 bis 15    | belegt ihn mit der Programmpriorität |
| 16          | ist frei (räumbar)                   |

#### 4.3 Platzwechsel

Ein Laufbereich steht den ihm zugeordneten Programmen abwechselnd zur Verfügung. Den Vorgang des Wechselns von PRP im Laufbereich (= Platzwechsel) führt das ORG selbsttätig aus.

Platzwechselanforderungen und -bedingungen bilden die Kriterien für die Wechselentscheidung des ORG.

---

Platzwechsel- Die Forderung eines PRP, in einen Laufbereich eintransfe-  
anforderungen riert zu werden, entsteht bei:

- Start oder Fortsetzen eines PRP
- Fortsetzen nach Ablauf der Zeitscheibe
- Anforderung des ORG zur HSP-Präsenz des PRP

Anforderungen führen erst zum Platzwechsel, wenn die Wechselbedingungen erfüllt sind. Die Anforderungen stehen nach Priorität und Zeit in einer Warteschlange.

Platzwechsel- Das Festhalten oder Freigeben des Laufbereichs durch ein  
bedingungen PRP ist ausgedrückt im Räumgewicht des Laufbereichs (s. 4.2). Ist die Anforderung höherprior als das Räumgewicht, kommt es zum sofortigen Platzwechsel. Dazu kommt es, wenn entweder eine höherpriorie Anforderung hinzu kommt oder wenn das Räumgewicht steigt (s.u.).

Räumgewicht = 0 (gesperrt, Sperrzähler) entsteht durch Aufruf "Platzwechsel sperren", durch E/A-Operation und durch ORG-internes Sperren.

Das Aufheben solcher Sperrungen läßt das Räumgewicht auf 5 bis 15 (Belegung nach Priorität) bzw. auf 16 steigen.

Räumgewicht = 16 (frei) entsteht durch PRP-Ende, Warten, Ablauf der Zeitscheibe und durch Aufruf "Platzwechsel anfordern".

Rücktransfer Der Rück- oder Austransfer bei Platzwechsel kann das ganze Programm oder nur den V-Teil beinhalten (Generierwunsch). In manchen Fällen unterbleibt der Rücktransfer.

PRP- Segmentierte Funktionen im ORG und Platzwechsel der PRP  
Koordinierung bedingen, daß Aufrufbearbeitungen unterbrechbar ablaufen müssen. Deshalb kann die Reihenfolge der Ausführungsschritte von der der Aufrufe abweichen (Überholvorgänge). Der Benutzer sollte dies bei der Programmkoordinierung innerhalb seines Anwendersystems und bei Änderungen der Programmarten (HRP <---> PRP) bedenken.

Ein-Ausgabe- Das ORG sperrt den Austransfer des PRP zwischen Operations-  
Operationen anstoß und Aufrufendebearbeitung (Sperrzähler). Bei daten-  
gepufferten E/A-Operationen wird nicht gesperrt (s. auch 5.4). Wartefunktionen ändern die Verhältnisse nicht.

Der Anwender kann eine ungepufferte E/A-Operation ausdrücklich ohne Laufbereichssperrung wünschen. Parameterblock, GEDA-Block und Datenfeld liegen dann außerhalb des PRP (jedoch HSP-resident; bei P-PRP in eingebundenen CD; bei S-PRP in Bereichs-CD, die nicht einbindbar sein müssen; s. 5.2).



---

#### 4.4 Koordinierungsfunktionen

Zusammen-  
fassung

Zur Koordinierung zwischen den Programmen eines Systems stellt das ORG vielseitig verwendbare Funktionen zur Verfügung, die der Anwender durch entsprechende Aufrufe benutzen kann.

Die Aufrufe zum Starten, Beenden, Unterbrechen, Anhalten und Fortsetzen haben zum Ziel, den Programmzustand zu ändern, wobei das koordinierte Programm passiv sein kann, weil ein anderes Programm oder das ORG die Funktion anstößt. Alle Funktionen können bei PRP Platzwechsel auslösen (s. 4.3).

Aufrufe zur Programmkoordination können folgende Modifikationen erhalten:

- Datenübergabe: Wenn ein Programm ein anderes Programm fortsetzt, kann es mit diesem Daten austauschen (s. 4.8.2).
- Zeitmodifikation: Ein Aufruf wird nicht zum Zeitpunkt der Abgabe, sondern zu einer vom Anwender angebbaren Zeit ausgeführt (s. 8).

##### 4.4.1 Programm starten und beenden

**Programmstart** Der Start eines Programms (durch Aufruf oder Kommando) entspricht dem Zustandswechsel dieses Programms von "ruhend" nach "ablauffähig" zuzüglich dem Einstellen der PT. Das ORG puffert keine Start-Aufrufe; dies ist mit Koordinierungszähler-Funktionen zu simulieren (s.u.).

Voraussetzung ist ein ruhendes Programm ohne Restaktivitäten aus vorausgegangenem Lauf. Neben dem Aufruf steht ein Kommando zur Verfügung. Parameter sind Objektidentifikation, Priorität und Zeitscheibeneinstellung.

**Modifikationen** Der Programmstart ist zeitverzögert und zyklisch ausführbar. Beim Startaufruf können Daten vom aufrufenden an das zu startende Programm übergeben werden.

**Startsperre** Der Start eines Programms läßt sich durch Aufruf oder Kommando sperren und wieder freigeben. Der Start für ein gesperrtes Programm wird zurückgewiesen.

- 
- Programmende** Das Beenden eines Programms entspricht dem Übergang in den Zustand "ruhend". Ein Programm kann sich selbst beenden oder kann von einem anderen Programm oder bei Fehlern vom ORG beendet/abgebrochen werden.
- Bei jedem Programmende wird die Programmnummernliste aktualisiert und Einträge gelöscht. Das ORG führt laufende Transfers zu Ende.
- Ende eigenes Programm** Vor Abgabe dieses Aufrufs sollen alle Programmaktivitäten abgeschlossen sein. Ein logisch korrektes Programmende ist sicherzustellen. Bei PRP wird mit oder ohne Rücktransferanforderung beendet.
- Ende fremd** Durch Aufruf oder Kommando wird das zu nennende Programm unabhängig von seinem Zustand sofort beendet, d.h. in den Zustand "ruhend" gesetzt. Zeitmodifikationen sind möglich. Ein Rücktransfer wird nicht angefordert, aber ein Eintransfer für einen Wiederstart.

#### 4.4.2 Programm unterbrechen und wieder fortsetzen

- Selbst anhalten, fortsetzen lassen** Ein Programm hält sich selbst an, es wartet auf die Fortsetzung durch ein anderes Programm. Hierzu steht ein Aufrufpaar zur Verfügung.

Zustandswechsel: ablauffähig in wartend bzw. wartend in ablauffähig (s. 4.2).  
Der Fortsetz-Aufruf kann auch zeitmodifiziert ausgeführt werden. Ferner ist bei dem Aufrufpaar Datenübergabe möglich.

Anmerkung: Diese Funktion kann auch durch Koordinierungszähler realisiert werden (s. 4.4.4).

- Anhalten, fortsetzen lassen** Mit einem Aufrufpaar kann ein fremdes (anzugebendes) HRP angehalten und später wieder fortgesetzt werden.  
Zustandswechsel: ablauffähig in angehalten oder wartend in angehalten-wartend bzw. angehalten in ablauffähig oder wartend-gehalten in wartend. Das angehaltene HRP merkt nichts davon; die Haltestelle ist unbekannt und beliebig. Beide Aufrufe können zeitverzögert gegeben werden.

#### 4.4.3 Auf Operationsende warten

**Operationen** Operationen von langer oder unbekannter Dauer werden durch einen Aufruf angestoßen und laufen, durch das ORG koordiniert, simultan zum anstoßenden Programm ab. Das Ende der Operation bildet meist ein Ereignis/Signal, worauf das ORG mit einer Routine die Operation abschließt (Aufrufendebearbeitung). Auf diese Weise koordiniert werden: Ein-/Ausgaben, Alarmbearbeitung, Dateizeigerbehandlung, Bedienungseingaben, Laden/Löschen, Testunterbrechungen, Monitorereignisse und Gerätezugriffe zum Steuern und Parametrieren.

Der Operationsaufruf wird bei der Abgabe durch Setzen des Tätigbits im Parameterblock als "tätig" erklärt, bis die Aufrufendebearbeitung das Tätigbit wieder löscht.

**Warten** Zur Steuerung des Programm-Ablaufs ist es häufig notwendig, daß das Programm auf das Ende von Operationen oder allgemein auf den Eintritt bestimmter Ereignisse warten muß.

Die Wartefunktion wird durch Aufrufe oder implizit angestoßen. Dabei prüft das ORG, ob das erwartete Ereignis bereits eingetroffen ist, und setzt entweder das Programm sofort fort (wenn das Ereignis bereits eingetroffen ist) oder unterbricht es (setzt es "wartend"). In diesem Fall wird das Programm in eine zugeordnete Warteschlange eingetragen. Beim Eintreffen des erwarteten Ereignisses sorgt die Operationsendebearbeitung für die Fortsetzung des Programms.

Für das Warten gibt es folgende Möglichkeiten:

- . Implizites Warten Der Operationsaufruf erhält die Modifikation "mit Warten". Das aufrufende Programm wird so lange unterbrochen, bis die Operation abgeschlossen ist; es läuft nicht simultan zur Operation.
- . Explizites Warten Hierbei gibt das Programm, das den Operationsaufruf gegeben hat, zu einem beliebigen späteren Zeitpunkt einen eigenen Warteaufruf. Er enthält als Parameter die Adresse des Parameterblocks des Operationsaufrufes. Damit ist bis zum Warteaufruf der Weiterlauf des Programms simultan zur Operation möglich.
- . Mehrfach-Warten Die Funktion des Mehrfachwartens bietet zwei Anwendungsmöglichkeiten:
  - Ein anderes Programm als dasjenige, das den Operationsaufruf gegeben hat, kann auf den Abschluß des Aufrufs warten.
  - Ein Programm kann auf die Beendigung einer von beliebig vielen, simultan ablaufenden Operationen warten. Es wird fortgesetzt, wenn eine dieser Operationen abgeschlossen ist.

Das Mehrfachwarten wird über die Koordinierungszählerfunktionen realisiert (s. 4.4.4) und läuft folgendermaßen ab:

- Ein Anwenderprogramm gibt einen eingerichteten Koordinierungszähler im Operationsaufruf an.
- Als Warteaufruf muß der Aufruf "Koordinierungszähler erniedrigen" mit der Modifikation "datenempfangsbereit" gegeben werden. Dieser wirkt dann wie der Warteaufruf beim explizitem Warten. Beim Fortsetzen werden dem Programm in zwei Registern Informationen übergeben: in R5 der Parameter "Fortsetzkennung" und in R6 die Parameterblockadresse des Operationsaufrufs.
- Das ORG bewirkt das Fortsetzen durch das Erhöhen des Koordinierungszählers in der Aufrufendbearbeitung.

. **Tätig-Bit-Abfrage**      Ferner besteht die Möglichkeit, durch Abfrage des "Tätig-Bits" im entsprechenden Parameterblock zu prüfen, ob eine Operation abgeschlossen ist.

#### 4.4.4 Koordinierungszähler

Zur Abwicklung der Aufgaben der Programmorganisation steht neben den festen Buchführungselementen für die Ablaufobjekte noch eine weitere Software-Einrichtung zur Verfügung: der Koordinierungszähler.

**Konzept**      Koordinierungszähler sind vom Anwender unter einem Namen einzurichten und werden vom ORG in einer Liste verwaltet. Sie bieten universelle, anpassungsfähige Möglichkeiten, um den Verkehr mehrerer Programme untereinander zu steuern etwa wegen der Benutzung eines gemeinsamen Objektes (Gerät, Datei, Programm).

Der Koordinierungszähler stellt eine Warteschlange dar, deren Abarbeitung durch eine Zählvariable mit zugehörigen Grenzwerten gesteuert wird. Warteschlangeneintragungen können nach Priorität oder nach Zeit erfolgen.

Die Zählvariable wird, von einem Anfangswert ausgehend, im Betrieb jeweils um 1 erhöht oder erniedrigt. Zwei Grenzwerte - die obere und untere Grenze ( $oG$ ,  $uG=0$ ) - teilen den Wertebereich der Zählvariablen  $V$  in drei Abschnitte:

$$V \leq 0 \quad 0 < V \leq oG \quad oG < V.$$

Der Anwender legt beim Einrichten die obere Grenze und den Anfangswert fest.

**Aufrufe**

- Koordinierungszähler einrichten (auch als Kommando)
- Koordinierungszähler löschen (auch als Kommando)
- Koordinierungszähler erhöhen
- Koordinierungszähler erniedrigen.

Wirkungsweise Gibt ein Programm einen Aufruf "Erhöhen" bzw. "Erniedrigen", so werden abhängig von der Veränderungsrichtung und dem Wert der Zählvariablen unterschiedliche Funktionen ausgelöst (s. Bild 4.4.4/1).

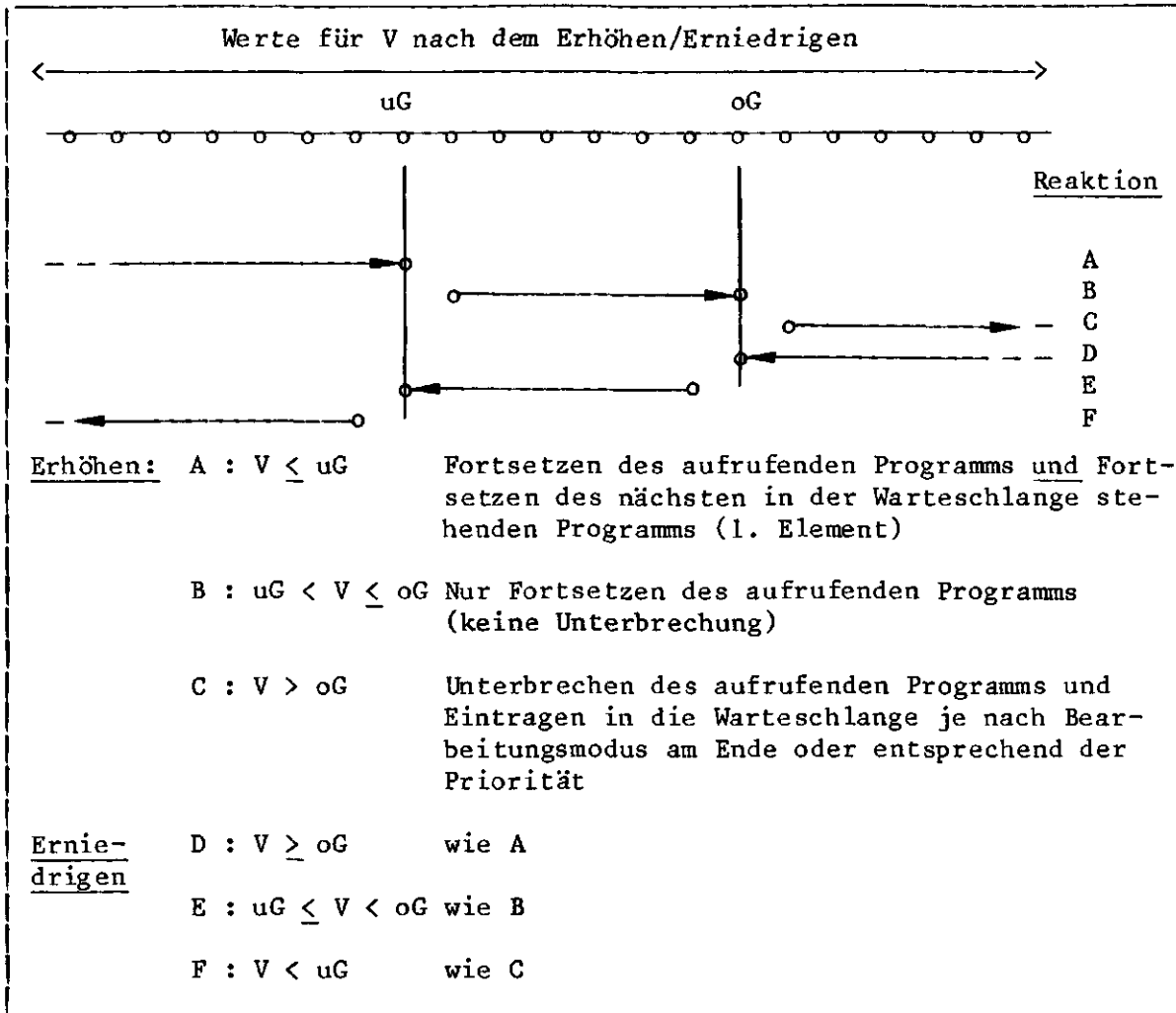


Bild 4.4.4/1: Wirkungsweise des Koordinierungszählers beim Erhöhen/Erniedrigen.

Modifikationen In Verbindung mit den Aufrufen zur Betätigung des Koordinierungszählers sind Datenübergabe (s. 4.8.2) und Zeitmodifikationen (s. 8) möglich.

Anwendungshinweise Mit diesem vielseitigen Koordinierungsinstrument ergeben sich u.a. folgende Möglichkeiten:

- Mehrfachwarten; hier beteiligt sich das ORG an der Koordinierung (Erhöhen mit den Funktionen A oder B; s. auch 4.4.3)
- Gepufferter Start; durch Funktion A oder B, Selbstbeendigung durch E oder F
- Belegen eines Betriebsmittels; durch Funktionen B oder C und Freigeben durch D oder E
- Warten auf Ereignisse; Ereignisse durch Funktionen A oder B; Warteaufruf durch E oder F.

Wegen dieser freizügigen Möglichkeit der Koordinierung von Programmen, die das ORG nicht überwachen kann, ist der Anwender für den Ablauf selbst verantwortlich. Logische Fehler und mangelnde Anzeigenabfragen können zu Blockierungen und anderen Fehlläufen führen.

#### 4.5 Programm- und Platzwechselfunktionen

Das ORG stellt Aufrufe/Funktionen zur Verfügung, die den Programmwechsel und den Platzwechsel direkt beeinflussen und die die Verweilzeit eines PRP im Laufbereich begrenzen.

- |                              |   |
|------------------------------|---|
| Programmwechsel<br>anfordern | Ein Programm kann für sich durch einen Aufruf eine Programmwechselstelle definieren. Das ORG entzieht dem Programm den Zentralprozessor und trägt es wieder in die Prioritätswarteschlange ein. Es erfolgt kein Zustandswechsel. Der Aufruf ermöglicht das gesteuerte wechselseitige Ablaufen von gleichprioreren Programmen. |
| Platzwechsel<br>sperrern     | Der Aufruf dient zum Setzen und Löschen einer Platzwechsel Sperre für das aufrufgebende Programm im Laufbereich. Eine Räumung wird bis zum Löschen der Sperre oder Programmende verhindert. Die Funktion benutzt den Sperrzähler.   |
| Platzwechsel<br>anfordern    | Der Aufruf erklärt für das aufrufgebende Programm im Laufbereich eine Räumbarkeitsstelle (Platzwechselanforderung). Ein Platzwechsel findet statt, wenn die Platzwechselbedingung dies zuläßt.  |
| Zeitscheiben-<br>bearbeitung | Die Zeitscheibenbearbeitung ermöglicht eine annähernd gleichmäßige Bearbeitung von gleichprioreren Programmen (ohne Wirkung auf Programme anderer Priorität). Die ORG-Zeitbearbeitung verwendet hierzu den Periodenwecker des Zeitimpulsgebers.   |

Zwei Arten von Begrenzungen stehen zur Verfügung:

- . CPU-Zeitscheibe Zum Programmwechsel. Die Belegung des Zentralprozessors durch Programme (HRP und PRP) wird begrenzt. Eine Zuteilungsdauer (= Zeitscheibe) von 250 ms ist für alle Programme voreingestellt. Per Aufruf/Kommando kann die Zeit verändert, die CPU-Zeitscheibenbearbeitung aus- und wieder eingeschaltet werden. Beim Starten oder mit einem eigenen Aufruf/Kommando können Programme der Kontrolle der CPU-Zeitscheibe unterstellt und ihr auch wieder entzogen werden.

War ein unterstelltes Programm die angegebene Zeit ununterbrochen aktuell, so wird ihm der Zentralprozessor entzogen bei gleichzeitigem Wiedereintrag in die Prioritätswarteschlange (Funktion wie "Programmwechsel anfordern", s.o.). So können alle anderen gleichprioreren ablauffähigen Programme weiterlaufen, ehe das Programm wieder aktuell wird. Existieren keine anderen Programme, bleibt das Programm für eine weitere Zeitscheibe aktuell.

AE 1

---

- PRP-Zeitscheibe Zum Platzwechsel. Die Zeit, wie lange ein PRP den Laufbereich ununterbrochen ablauffähig oder wartend belegen darf, wird programmspezifisch begrenzt. Die Verweilzeitvorgabe erfolgt zweistufig. Eine PRP-Zeitscheibe ist mit 1 s voreingestellt; sie kann per Aufruf/Kommando abgestellt (keine PRP-Zeitscheibenbearbeitung) und wieder eingestellt werden.

Beim Starten oder mit einem Aufruf/Kommando gibt der Anwender programmspezifisch die Verweilzeit an durch die Anzahl von PRP-Zeitscheiben und schaltet damit die Überwachung für dieses Programm ein. Ebenso ist programmspezifisches Ausschalten möglich.

Hat ein PRP seine Verweilzeit im Laufbereich verbraucht, räumt das ORG den Laufbereich, stellt für das PRP aber wieder eine prioritätsgerechte Platzwechselanforderung (Funktion wie "Platzwechsel anfordern", s.o.). Dadurch kommen gleichpriorere anfordernde PRP zum Zuge. Ohne zu Räumen teilt das ORG eine weitere Zeitscheibe zu, wenn keine gleichprioreren Anforderungen vorliegen, wenn das PRP implizit oder explizit gesperrt ist oder wenn das PRP in der abgelaufenen Zeitscheibe nicht ein einziges Mal aktuell war.

#### 4.6 Strukturierung und Benutzung der alternativen Laufbereiche

Die alternativen Laufbereiche (s. 2.6) können für ein Spektrum von Programmen den Bearbeitungsdurchsatz gegenüber fest zugeordneten Laufbereichen erhöhen, zumal Anzahl und Längen online anpaßbar sind. Der Anwender kann alle oder einen Teil der Laufbereiche als alternative Laufbereiche generieren (s. 2.6).

- Auswahl eines Alternativlaufbereichs Beim Laden bestimmt der Anwender, daß das Programm in einem Alternativlaufbereich ablaufen soll. Bei jedem Eintransfer wählt das ORG neu unter allen alternativen Laufbereichen einen passenden aus. Es sucht aus den belegbaren den kürzesten, der für das PRP ausreicht. Gemäß Anforderungswarteschlange für Alternativlaufbereiche wird nicht nur versucht, das höchstpriorere PRP sondern auch andere PRP unterzubringen. Ein kürzeres niederprioreres PRP kann durchaus einen Laufbereich finden; Prioritätsüberholungen können stattfinden. Alle anderen Platzwechselmechanismen sind unverändert.

- Umstrukturierung Unter Struktur versteht man die Anzahl und die Längen der Laufbereiche, die zusammen einen geschlossenen HSP-Bereich belegen (s. Bild 1/2). Der Anwender generiert eine Struktur, die mittels Aufruf/Kommando auch geändert werden kann (Grundstruktur).

- 
- . Angaben Mit einem Aufruf gibt der Anwender die Anzahl und Längen für eine neue Struktur an sowie eine Zeit, in der die Umstrukturierung realisiert sein muß (s.u.). Die Summe der Längen und die Anzahl dürfen die Grenzwerte (beim Generieren vorgegeben) nicht übersteigen. Bei der Disposition sind auch die noch ablauffähigen bzw. wartenden PRP und die System-PRP zu berücksichtigen.
- . Vorgang Das ORG räumt zuerst die alternativen Laufbereiche; bei gesperrten Laufbereichen wartet das ORG, bis sie nicht mehr gesperrt sind, und räumt sie dann. Ist dies nicht in der vorgegebenen Zeit möglich, wird der Aufruf mit Anzeige abgebrochen und der PRP-Betrieb wieder aufgenommen (s.u.).
- Nach erfolgreichem Räumen strukturiert das ORG in die neue Grundstruktur um. Danach nimmt es wieder den PRP-Betrieb auf durch Neu-Auswahl von Laufbereichen für die Anforderungen.
- temporäre Umstrukturierung Erhält der Anwender (meist beim Starten oder Laden) die Anzeige, daß es keinen genügend großen Alternativlaufbereich gibt, dann kann er durch einen besonderen Aufruf veranlassen, daß so umstrukturiert wird, daß das betreffende (im Aufruf zu nennende) PRP ablaufen kann. Ein weiterer Aufruf ermöglicht das Zurück-Strukturieren auf die Grundstruktur, z.B. dann, wenn das lange Programme abgelaufen ist. Strukturierungsvorgang s.o.
- Struktur-  
auskunft Der Anwender kann sich mittels Aufruf/Kommando die zur Zeit aktuelle Struktur mitteilen lassen.
- Sperren eines PRP im alternativen Laufbereich Ein besonderer Aufruf erlaubt das Sperren eines PRP (Sperrzähler) in einem alternativen Laufbereich, sobald es wieder eintransferiert ist. Den Laufbereich wählt das ORG aus (s.o.), oder der Anwender nennt ihn (s.u.). Die Sperrung wird mittels Aufruf oder bei Programmende aufgehoben. Das ORG weist Umstrukturierungen zurück, wenn alternative Laufbereiche derart gesperrt sind.
- Fixieren eines PRP auf einen alternativen Laufbereich Mit einem Aufruf kann der Anwender für ein PRP einen bestimmten alternativen Laufbereich wählen (keine Auswahl des ORG). Die Fixierung wird mittels Aufruf oder bei Programmende aufgehoben. Existieren fixierte PRP, weist das ORG Umstrukturierungen zurück.
- Nummern der alternativen Laufbereiche Normalerweise muß ein Alternativlaufbereich nicht angesprochen werden. Das ORG vergibt für die alternativen Laufbereiche die Nummern n+1, n+2,... in der Reihenfolge der Längenvorgaben, wobei "n" die höchste Nummer der festen Laufbereiche ist.
- aktuelle  
Tafelzeiger Das ORG übergibt bei jedem Laufbereichswechsel (Zuteilung eines neuen alternativen Laufbereichs) den neuen Tafelzeiger an das Anwenderprogramm. Dieses hat nur die Übergabeadresse in die PT Wort 13 einzutragen. Ist in Wort 13 nichts eingetragen (Wort 13 = 0), unterbleibt diese ORG-Funktion.



---

#### 4.7 Parameterübergabe, Objektzugriffe

Diese Funktionen dienen dazu

- dem Anwender bestimmte Objektparameter für Koordinierungszwecke zugänglich zu machen und
- den lesenden oder schreibenden Zugriff zu Objekten zu unterstützen.

Es stehen Aufrufe zur Verfügung, teilweise auch Kommandos. Die zu übergebenden Informationen trägt das ORG in den jeweiligen Parameterblock ein bzw. druckt sie aus.

|                               |  |
|-------------------------------|--|
| Programmparameterübergabe     | Folgende Parameter aus Parametertafel und Programmnummernliste werden übergeben: Objektnummer, Objektart, Programmzustand und -Modifikationen, HSP-Präsenz, Speicherort, Priorität, ORG-Kennung und Meldegerät.  |
| Testparameterübergabe         | Übergabe etwa wie bei Programmparameter, zusätzlich Tafelzeiger und virtuelle Objektadressen.  |
| Common-Data-Adresse ermitteln | Der Aufruf übergibt die virtuelle Anfangsadresse von Common Data und (reentranten) Common Code. Er bewirkt bei Bereich-CD und CC das Einbinden der ÜT des Commonbereiches bzw. der CC-ÜT in die ÜT des rufenden Objektes. Das Ankoppeln von CC muß beim Laden erfolgen.  |
| Objektadresse ermitteln       | Der Aufruf ist für den Zugriff zu Objekten (HRP, CD, CC, ORG) in einem fremden Adreßraum vorgesehen, läßt sich aber auch im eigenen Adreßraum verwenden. Das ORG übergibt die virtuelle Anfangsadresse des gewünschten Objektes und stellt das Tafelzeigerregister TZR2 auf die Übersetzungstafel des gerufenen Objektes. Weiter wird mitgeteilt, ob das Objekt ORG, Programm, CC oder CD ist und ob Tafelzeigerregister TZR1 ungleich TZR2 ist.<br><br>Der Aufruf bewirkt im Unterschied zur CD-Adreßermittlung (s.o.) kein Einbinden des Commonbereichs. Mit dem Aufruf läßt sich TZR2 wieder auf den eigenen Adreßraum zurückstellen, er ist in diesem Fall auch für PRP erlaubt. |
| ORG-Daten übergeben           | Dieser Aufruf stellt folgende ORG-Daten in einem Hilfsfeld zur Verfügung: <ul style="list-style-type: none"><li>- ORG-Kennung, Programmnummernliste, Geräteliste,</li><li>- Geräteverweise, Ausfall- und Datumswechselzeit, HSP-Ausbau</li><li>- Anzahl von Objekten und Geräten</li><li>- Systemkennzeichen, Schnittstellenadressen,</li><li>- Nummer des Wiederanlaufprogramms u.a.</li></ul>  |

Objekt-  
zugriffe

Als zugreifendes Objekt wird stets ein Programm (HRP, PRP) angenommen. Gerufene Objekte sind HRP, CC und CD (ÜT-gleich oder ÜT-fremd), PRP und ORG. Sind die Objektnummern nicht bekannt, können sie über den Objektnamen beschafft werden (Aufruf Namen-Nummern-Zuordnung).

Zugriff bedeutet das Adressieren von Datenzellen oder Feldern zwecks Datenaustausch: Daten und Blöcke aus anderen Objekten sind in das Programm zu übernehmen oder wieder abzuspeichern; in andere Objekte ausgelagerte Felder und Blöcke (z. B. PB, GEDA) müssen ständig adressierbar sein.

Zugriff bedeutet aber auch das Anspringen von Befehlsfolgen (Unterprogrammen) in anderen Objekten.

Zugriffe erfolgen als:

- Lade- und Speicherbefehle
- Feldübertragungsaufrufe (s. 4.8.3)
- Sprungbefehle in Fremdobjekte (Unterprogramm- und Rücksprünge)

Für den Zugriff sind (je nach Objekt, Adreßraum und Zugriffsart) als Vorbereitungen nötig:

- Adreßbeschaffung; virtuelle Adresse oder aus Objektanfangs- und Relativadresse zusammengesetzt (Aufrufe: Common Data- oder Objekt-Adresse ermitteln, Datenübergabe)
- Ladebinden mit Adreßbeschaffung (Externaufrufe) und ÜT-Einbindung (s. 12)
- TZR2-Einstellung bei Zugriff auf Daten (Lade- oder Speicherbefehle) in einem fremden Adreßraum, s. /26/ (Aufruf: Objektadresse ermitteln) und zusätzlich
- Adressierungsweichenregister-Einstellung (AWR).  
Im Lade/Speicherbefehl hat die Adreßangabe die Form: (adr + Rx). Nun ist im AWR (Spezialregister R27) jedem der Bits 0 bis 15 eines der Standardregister R0 bis R7, C0 bis G7 zugeordnet. Ist ein AWR-Bit x = 1, so wird die Adresse: (adr + Rx) über TZ2 im fremden Adreßraum gesucht; ist Bit x = 0, so wird über TZ1 im eigenen Adreßraum zugegriffen. Beispiel:  
R27 := 'H=0080'      Bit 8 = 1, Bit 7 = 0  
R5 := (A + G0)      Lesen über TZ2 aus fremdem Adreßraum  
(B + R7) := R5      Schreiben über TZ1 in eigenen Adreßraum
- Sprünge in einen fremden Adreßraum nur im Rahmen der Adreßraumerweiterung für Anwenderprogramme möglich (Auslagerung von Subsystem-Code, Schnittstellen über normierten Adapter, Umstellen des TZR3 durch Adreßraumbefehle, s. /216/)

#### 4.8 Datenaustausch zwischen Programmen

Übersicht

Der Datenaustausch zwischen Programmen kann erfolgen:

- über Lade- und Speicherbefehle bei gemeinsamem Adreßraum,
- über dynamisch vom ORG verwaltete Feldbereiche in Feldeinheiten,
- über Dateien (s. 6),
- über Common Data (s. 4.8.1),
- durch Datenübergabe in 2 definierten Registern (s. 4.8.2),
- durch Feldübertragungen zwischen Programmen/Common Data/Common Code (s. 4.8.3),
- durch programmierte Bedienung (s. 9.5).

#### 4.8.1 Datenaustausch über Common Data

Common Data (s. 1.2 und 2.4) können in Paketen oder im Commonbereich stehen, haben eine Objektnummer, aber keine eigene Parametertafel und sind durch Laden einzurichten (s. 12). Anfangsadresse (und Länge) eines CD kann der Benutzer über Aufrufe (s. 4.7) ermitteln und damit den Zugriff programmieren. Reentrant geladene Common Codes (s. 2.4) können wie Common Data behandelt werden.

**Zugriff** Möglich sind Lesen und Schreiben durch Lade- und Speicherbefehle, Feldübertragungsaufrufe (s. 4.8.3) und Sprünge (Sprungbefehle). Datenfelder und (PB/GEDA-)Blöcke für E/A-Aufrufe können in CD angesprochen werden. Für die Adressierung in Verbindung mit dem Zugriff ist zu unterscheiden zwischen CD im eigenen Adreßraum oder im fremden Adreßraum. Die mit den Aufrufen mitgeteilten Adressen sind stets virtuell und daher erst durch Bezug auf einen Tafelzeiger (TZ1 oder TZ2) gültig.

**CD im eigenen Adreßraum**

- Stehen CD im gleichen Paket wie das rufende Programm oder sind Bereichs-CD bereits eingebunden (ÜT übernommen), so sind für die vorgenannten Zugriffsarten keine Besonderheiten zu berücksichtigen (symbolische Adressen für Externaufrufadressen).
- Steht CD im noch nicht eingebundenen Commonbereich, so bewirkt der Aufruf "CD-Adresse ermitteln" das Einbinden des Adreßraums des Commonbereiches in den Adreßraum des rufenden Programms. Danach Zugriffe wie oben. Ist der gerufene Bereichs-CD nicht einbindbar, erfolgt der Zugriff nach den Regeln für fremde Adreßräume (s.u.).

**CD im fremden Adreßraum** Dem Zugriff muß der Aufruf "Objektadresse ermitteln" vorausgehen (s. 4.7). Ein-Ausgabe-Datenfelder in solchen CD sind nur über indirekte Adressierung (Adreßliste) zugänglich (s. 21.3.10 in /101/). Für den Zugriff mit Lade- und Speicherbefehlen sind nur die Register zulässig, die zuvor an den entsprechenden Bitstellen des Adressierungsweichenregisters (AWR) für den Gebrauch im zweiten Adreßraum markiert worden sind.

#### 4.8.2 Datenübergabe in Registern

**Senden, Empfangen** Die Übergabe von Daten in den Registern R5 und R6 ist in Verbindung mit Koordinierungsaufrufen möglich. Das sendende Programm muß dies im Aufruf wünschen, ebenso das empfangende Programm im korrespondierenden Aufruf.

**Aufrufe** Folgende Aufrufe ermöglichen Senden und/oder Empfangen:

|                                  |                          |
|----------------------------------|--------------------------|
| Programm starten                 | mit Senden               |
| Auf Fortsetzung warten           | mit Empfangen            |
| Programm fortsetzen              | mit Senden               |
| Koordinierungszähler erhöhen     | mit Senden und Empfangen |
| Koordinierungszähler erniedrigen | mit Senden und Empfangen |

Die Aufrufe können auch zeitverzögert oder zyklisch ausgeführt werden. Beim Starten sind Besonderheiten zu beachten. Datenübergabe bei Warten und Mehrfachwarten s. 4.4.3.

---

### 4.8.3 Datenaustausch durch Feldübertragung

**Aufrufe** Mit den Feldübertragungsaufrufen (Lesen, Schreiben) ist Datenübertragung zwischen beliebigen Programmen (HRP und PRP), Paketen, CC und CD in beiden Richtungen und auch der Zugriff zum ORG möglich.

Die Daten werden vom angegebenen Sendefeld zum Empfangsfeld übertragen, im binären Modus bis zum Ende des kleineren Feldes, im alphanumerischen Modus höchstens bis zum Endezeichen (ETX, ETB).

**Lage des Sende/Empfangsfeldes** Das aufrufgebende Programm ist mit seinem Feld zwangsläufig HSP-präsent.

Ist das andere (= passive) Objekt ein HRP, CC oder CD, ist auch sein Feld HSP-präsent. Existiert ein Abbild, entscheidet der Anwender durch einen Aufrufmodus, ob mit dem Abbild und/oder dem HSP verkehrt wird.

Ist das andere Objekt ein PRP, so sind für sein Feld zwei Stellen zu berücksichtigen: der Laufbereich und der Wartebereich. Abhängig von der momentanen PRP-Präsenz und dem Aufrufmodus wird mit dem Wartebereich und/oder dem Laufbereich verkehrt.

Feldübertragungen zwischen segmentierten Programmen sind nur eingeschränkt möglich.

**Geschützte Bereiche** Lesen aus geschützten und Schreiben in geschützte Seiten (Schutzbit in der ÜT) ist bedingt möglich.

**Koordinierung** Das ORG unterbricht Programme, die Feldübertragungsaufrufe abgeben, bis zum Ende der Übertragung. PRP im Laufbereich sind dabei nicht räumbar.

**Adressierung** In den Aufruf-Parameterblöcken zur Feldübertragung sind Sende- und Empfangsfeld als auf die dem Objekt zugeordnete ÜT bezogene virtuelle Adressen einzutragen.

Im eigenen Feld geschieht das mit den symbolischen Adressen. Die Angabe des fremden Feldes setzt eine Ermittlung der virtuellen Adresse voraus. Je nach Objekt und Adreßraum ist die Anfangsadresse des Objektes zu beschaffen (s. 4.7). Die Summe aus Objektanfangsadresse und Felddistanz vom Objektanfang ist die anzugebende Adresse des Fremdfeldes. Sind die Fremdadressen im aufrufenden Programm als Externaufrufe definiert und gebunden, können die symbolischen Adressen direkt benutzt werden.

Die Übertragung ist nur für ein zusammenhängendes Datenfeld möglich.

---

#### 4.9 Segmentierte Programme

Der Anwender hat folgende Arten von segmentierten Programmen zu unterscheiden:

segmentierte HRP      Rootsegment ständig im HSP, Segmente bei Ablauf abwechselnd im HSP, stückweise ablauffertig aufbereitet

segmentierte PRP      Rootsegment bei Ablauf im Laufbereich, Segmente bei Ablauf abwechselnd im Laufbereich, stückweise ablauffertig aufbereitet

SHP                      HSP-segmentiertes HRP: in ganzer Länge, d.h. alle Segmente ständig im HSP, stückweise ablauffertig aufbereitet

Segmentierung      Mit segmentierten Programmen lassen sich Beschränkungen des HSP-Platzes, der Laufbereichslängen und - bei SHP - des Adreßraumes umgehen. Erstellung und Ablauf segmentierter Programme sind durch nachstehende Arbeitsschritte gekennzeichnet:

- Strukturierung, keine Hilfsmittel
- Binden, mit Binder LNKR-M (s. /213/)
- Laden, mit Lader des ORG
- Ablauf, ORG-unterstützt

Strukturierung      Das Konzept eines segmentierten Programms legt der Anwender fest (s. Bild 4.9/1):

- Er zerlegt die Programmaufgabe in möglichst selbständig arbeitsfähige Teilstücke (Segmente, max. 255 möglich).
- Diese Segmente müssen übersetzte Einheiten sein; nach der Übersetzung werden sie zu Grundsprache-Moduln.
- Ein Rumpfstück (Rootsegment) des Programms enthält nicht segmentierbare Hauptteile sowie gemeinsame Datenfelder und steuert die Ablauffolge der übrigen Segmente.
- Die Struktur des zeitlichen und ablauftechnischen Zusammenwirkens aller Segmente - die Überlagerungs-Struktur - legt der Anwender fest und teilt sie dem Binder mit.

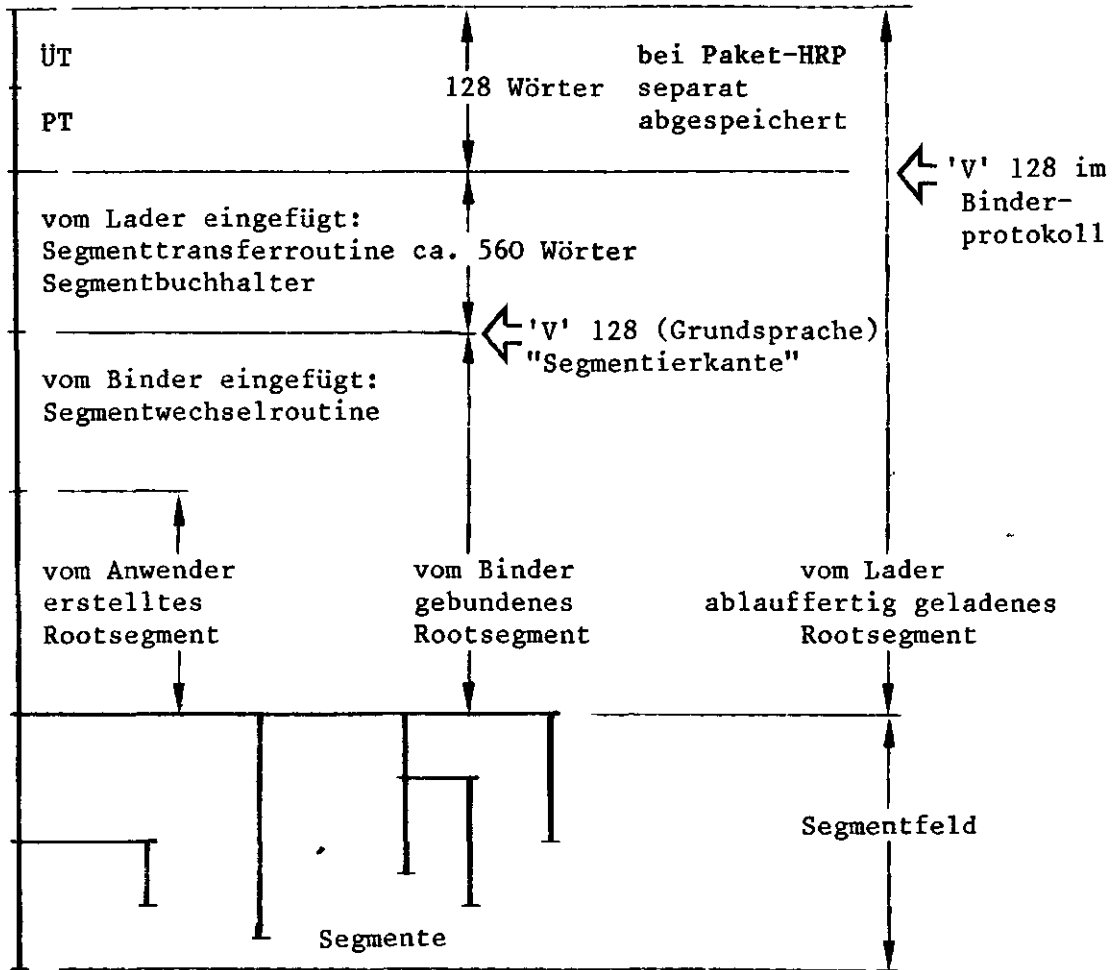


Bild 4.9/1 Überlagerungsstruktur segmentierter Programme

- Binden** Mit dem Binder lassen sich die in Grundsprache-Moduln vorliegenden Segmente zu segmentierten Programmen zusammenfügen. Wesentliche Bindefunktionen: Herstellung von Adreßbezügen zwischen den Segmenten gemäß Strukturbeschreibung (Binder-Steueranweisungen), Einbau der Segmentwechselroutine am Anfang des Rootsegmentes, Abspeicherung der gebundenen Segmente als Elemente einer Grundsprachebibliothek. Die Elemente enthalten nur V-Teile; I-Teile werden den V-Teilen zugerechnet. Alle Adreßpegel beziehen sich auf die Lage zum Anfang des Rootsegmentes, der die Adresse 'V'128 hat (Segmentierkante).
- Laden** Der Lader erkennt am Ladeobjekt ein Programm als segmentiert, richtet die Arbeitsformdatei ein, legt dorthin die Segmente ab und stellt eine Segmenttransferoutine mit zugehörigen Buchführungsdaten (Segmentbuchhalter) an den Anfang des Rootsegmentes. Danach überträgt der Lader das Programm von der Arbeitsform in den HSP bzw. Wartebereich (s. 12.3).
- Ablauf** Der Ablauf eines segmentierten Programms beginnt im Rootsegment. Alle nachfolgenden Befehle und ORG-Aufrufe im Rootsegment einschließlich der Segmentwechsel laufen unter der Regie des segmentierten Programms selbst ab; ebenso wie die Abläufe in und die Zugriffe zu den Segmenten.

---

## 5 Ein-/Ausgabe-Organisation

**Anwender-  
schnittstelle** Für den Verkehr mit peripheren Geräten stehen Standard-  
Aufrufe und Sonder-Aufrufe zur Verfügung. Standard-Aufrufe  
ermöglichen es dem Anwender, unabhängig von der Ausstattung  
einer Anlage, d.h. unabhängig von den für die Ein-/Ausgaben  
verwendeten Geräten und der Darstellung der Daten auf den  
peripheren Datenträgern, Programme zu erstellen.

Die Unterschiede der verschiedenen Geräte (Versorgung, physikalische Blockgrößen, Steuerfunktionen usw.) werden so weit als möglich vom ORG ausgeglichen. Der Anwender braucht deshalb auch nicht mit der physikalischen Struktur der Geräte vertraut zu sein. Die ein- oder auszugehenden alphanumerischen Daten müssen sich auf die allen Standard-Geräten gemeinsamen gleichartigen Zeichen (insbesondere Steuerzeichen) beschränken (s. 5.3).

Abweichungen von den Bedingungen verbieten nicht die Benutzung der Standard-Aufrufe, schränken aber u.U. die Geräteunabhängigkeit ein und sind vom Anwender zu berücksichtigen.

**Gerätegruppen** Aus der Sicht des ORG wird die Peripherie - unabhängig von der Einteilung durch die Hardware - in den Komplexen Standard-, Prozeß- und Sonderperipherie zusammengefaßt. Gerät im Sinne des ORG ist ein Gebilde, das im Rahmen der Geräte-  
liste in Form eines Geräteblocks beschrieben werden kann  
(logisches Gerät).

- . **Standard-  
peripherie** Standard-Aufrufe sind für Geräte der Standardperipherie vorgesehen. PSD werden im Datei- oder Geräteverkehr betrieben (s. 6). Für Magnetbandkassetten und Magnetbänder stehen Spezialaufrufe zur Verfügung. Datenübertragung ist mit eigenen Koppelaufrufen möglich (s. 5.3.3).
- . **Prozeß-  
peripherie** Die zweite Gerätegruppe stellt die Prozeßperipherie dar, mit deren Geräten sowohl das ORG als auch der Anwender über Makros verkehrt (s. 5.5).
- . **Sonder-  
peripherie** Alle nicht unter Standard- und Prozeßperipherie definierten Geräte sind als Sonderperipherie nicht Gegenstand der ORG-Beschreibung. Die Verantwortung des Geräteverkehrs über eine besondere Schnittstelle liegt beim Anwender (s. 5.6).

**Betriebs-  
hinweise** Bei Standard- und Sonder-Aufrufen ist Simultanarbeit zwischen Programmen und Geräten sowie der Geräte untereinander möglich.

Das ORG stellt für langsame Ein- und Ausgabegeräte der Standardperipherie eine Datenpufferung zur Verfügung (s. 5.4).

Bei allen Ein-Ausgabe-Aufrufen ist neben dem Parameterblock ein Geräte-Datei-Block (GEDA-Block) je Gerät bzw. Datei nötig.

---

## 5.1 Koordinierung des Ein-/Ausgabe-Verkehrs

Bevor der eigentliche Datentransfer ausgeführt werden kann, sind einige vorbereitende Maßnahmen notwendig (Dateiverkehr eingeschlossen).

|                            |   |
|----------------------------|---|
| Geräte-<br>Zuteilung       | Die Festlegung, mit welchem Gerät der E/A-Verkehr stattfinden soll, kann erfolgen durch Angabe des Geräts im GEDA-Block, programmiertes Eintragen des log. Gerätenamens in den GEDA-Block, programmiertes Eintragen des Geräteverweises oder Zuteilung über einen Bedienungsdialog. |
| Zuteilungs-<br>arten       | Der Anwender kann ein Gerät explizit, implizit und programmiert zuteilen.   |
| Geräte-<br>Belegung        | Die Belegung von Geräten durch bestimmte Programme kann mit dem Koordinierungszähler vorgenommen werden (s. 4.4.4). Das ORG bietet hierzu keine eigene Funktion.  |
| Programm-<br>Koordinierung | Weiterlaufen oder Warten des Programms s. 4.4.3<br>Laufbereichsbelegung s. 4.3  |

## 5.2 Standard-Ein-/Ausgabe

Mit Standard-Ein-/Ausgabe-Aufrufen kann der Anwender Programme geräteunabhängig und ohne Kenntnis der physikalischen Geräte-Eigenschaften erstellen.

|                                    |  |
|------------------------------------|--|
| Datenart                           | Der Anwender hat zwei Arten der Datendarstellung (Datenarten) zu unterscheiden:<br><br>alphanumerisch Byte-Darstellung von Texten entsprechend dem ISO-7-Bit-Code in 8-Bit-Umgebung, mit Steuerzeichen<br><br>binär beliebige Bitmuster im Hauptspeicher, ohne Steuerzeichen   |
| Aufruf-<br>Modifikation-<br>tionen | Die Modifikationen der Standard-Aufrufe betreffen (s. 21.3.10 in /101/): <ul style="list-style-type: none"><li>- Datenart (mit, ohne Steuerzeichenerkennung)</li><li>- Adreßangabe (direkt, indirekt)</li><li>- Adressierungseinheit (Wort, Byte)</li><li>- Adreßraum für Blöcke/Felder(eigen, fremd)</li><li>- Peripherieanzeigen-Übergabe</li><li>- Wartemodifikation (implizites Warten, Mehrfachwarten)</li><li>- Datenpufferung</li><li>- ohne Laufbereichssperrung</li></ul> |



Blöcke  
Datenfeld Die Bestandteile des Standard-E/A-Aufrufs (wie z.B. Parameterblock, GEDA-Block, Datenfeldliste, Datenfeld) stehen in der Regel im eigenen Adreßraum. Hierzu gehören auch eingebundene Adressen (aus CD).

Datenfelder und Blöcke (PB, GEDA-Block, Datenfeldliste) können auch in einem fremden Adreßraum (CD oder HRP) stehen. Die Blöcke sind nur bei folgenden E/A-Aufrufen auslagerbar:

- \$STEI/AU.., 21.3.10 bis .12 in /101/,
- dateiorganisatorische Aufrufe \$DA.., 21.4 in /101/,
- Magnetbandkassettensteuerung \$MKR.., 21.3.22 in /101/,
- Magnetbandaufrufe \$MB.., 1 in /106/.

(Das ORG erlaubt das Auslagern bei Aufrufen mit der Aufrufverschlüsselung S2 = 2, 19, 21, 47; s. 20.5 in /101/).

Zu einem Aufruf stehen die zugehörigen Blöcke (PB, GEDA-Block, evt. Datenfeldliste) in einem CD oder HRP und das evtl. zugehörige Datenfeld im selben Objekt oder in einem anderen Adreßraum (CD oder HRP). Angabemöglichkeiten und -formen s. 21.3.10.7. Somit ist ein Blöcke-CD denkbar, in dem alle PB, GEDA-Blöcke und Datenfeldlisten eines Programm(system)s stehen; entsprechend auch ein (oder dasselbe) CD für die Datenfelder.

Transferwarteschlange Zur Koordinierung des Verkehrs der Programme mit den Geräten baut das ORG für jede simultan arbeitsfähige Gerätefunktion (Gerät/Teilfunktion/Gerätegruppe) eine eigene Transfer-WS auf. In einer Transfer-WS stehen die Aufrufe der Programme nach Priorität sortiert.

Mit einem Sonderaufruf ist es möglich, in den Ablauf eines E/A-Aufrufs und der Gerätetätigkeit einzugreifen und die gesamte Transferwarteschlange eines Gerätes zu räumen.

E/A-Verkehr Der E/A-Verkehr wird teils vom ORG, teils von den Geräten simultan zu den Programmen abgewickelt und überwacht. Die Peripherieanzeigen werden ausgewertet; es kann zu vorzeitigem Abbruch des Verkehrs kommen. Bei bestimmten Gerätefehlern (s. 13) gibt das ORG je nach Generierangabe eine quittierbare Fehlermeldung aus.

Folgende Bedingungen kennzeichnen ein Transferende: Erkennen eines Endezeichens (alphanumerische Daten), Datenfeldende, Auftreten eines nicht korrigierbaren Fehlers (der zum Transferabbruch führt).

Die Endadresse ist die Byteadresse des ersten nicht mehr übertragenen Zeichens. Sie hat bei normal abgeschlossenem Transfer den um Eins erhöhten Wert der Datenfeldendadresse. Wurde der Transfer mittendrin bzw. gleich zu Beginn abgebrochen, zeigt der Wert auf die Stelle im Datenfeld bzw. auf die Datenfeldanfangsadresse. Kam es nicht zum Transfer, ist der Wert Null oder unverändert.

---

### 5.3 Standardgeräteverkehr

Das verfügbare Gerätespektrum der Standardperipherie ist in 1.2.5 vorgestellt worden. Die Geräte der Standardperipherie sind (ausgenommen Datenübertragungssteuerungen) off- und online schaltbar.

**Zeichenvorrat** Texte benutzen den sog. Standardzeichenvorrat, eine Teilmenge des ISO-7-Bit-Codes (gleich CCITT-Alphabet Nr. 5 und DIN 66003) mit der USASCII- bzw. ANSI-Belegung. Der Standardzeichenvorrat enthält 64 abdruckbare Zeichen, die für alle Geräte zugelassen sind, sowie die Steuerzeichen LF, FF, ETX und ETB.

Zeichen außerhalb des Standardzeichenvorrats werden stets bearbeitet, d.h. im HSP abgelegt oder ausgegeben, ggf. auch wirkungslos unterdrückt bzw. durch eine Lücke oder ein Schmierzeichen ersetzt.

**Steuerzeichen** Aus der Menge der Steuerzeichen des Codes sind hauptsächlich die vier Steuerzeichen LF, FF, ETB, ETX von Bedeutung. Die übrigen Steuerzeichen sind teilweise gerätespezifisch, teilweise in den Geräten nicht realisiert.

**Texte** Abhängig vom Umfang der erlaubten Steuerzeichen lassen sich verschiedene Arten des Textes unterscheiden:

**.geräteunabhängig** Der geräteunabhängige Text enthält nur die Steuerzeichen LF, FF und ETX. Texte mit diesen Zeichen erzeugen unter Berücksichtigung der kleinsten Zeilenlänge (Schreibstellenanzahl) auf allen textdarstellenden Geräten identische Schriftbilder.

**.bedingt geräteunabhängig** Der bedingt geräteunabhängige Text enthält zusätzlich das Zeichen ETB. Das Zeichen ETB dient u.a. dazu, mit ETX abgeschlossene Texte in mehrere Textblöcke zu unterteilen, die mit je einem Aufruf übertragen werden.

**.gerätespezifisch** Der gerätespezifische Text enthält außer den vier vorgenannten Steuerzeichen weitere, die jeweils gerätespezifisch definiert sind (z.B. Datensichtstation).

---

|                               |  |
|-------------------------------|--|
| Fehler-<br>reaktionen         | <p>Fehler der Hardware (Anschaltung, Bus, Gerät), die von der Hardware oder dem ORG erkannt werden, führen abhängig vom Gerätebaustein zu Aufrufanzeigen und Meldungen (s. 13). Einige Gerätefehler werden quittierbar gemeldet, wenn der Anwender dies so einstellt (beim Generieren oder durch Kommando /ERRORCHANGE s. /103/). Bei Plus-Quittung erfolgt Aufrufwiederholung oder Fortsetzung, bei Minus-Quittung wird der Aufruf mit Anzeigen abgebrochen. Die Zeit bis zur Quittung geht in die Bearbeitungszeit des Aufrufs ein.</p> <p>Bei AMBOSS 4 steuert das Bediensystem (s. /500/) die Meldungs-<br/>ausgabe und Quittungsbehandlung.</p>   |
| Zeit-<br>überwachung          | <p>Der Standardgeräte-Verkehr kann gerätespezifisch zeitüberwacht werden. Der Anwender wählt die Überwachungszeit in Minuten so, daß bei Ablauf der Zeit eine Gerätestörung zu vermuten ist. Das ORG bricht bei Ansprechen der Zeitüberwachung den E/A-Verkehr (mit besonderer Fehlermeldung) ab, um Systemblockierung zu vermeiden und den Verkehr für weitere Aufträge freizugeben. Beim Generieren ist eine Überwachung gerätespezifisch angebar. Kommandos erlauben das globale oder gerätespezifische Ein/Ausschalten der Zeitüberwachung; einschließlich Zeitvorgabe gerätespezifisch (/TIMER..., /103/).</p> <p>Nicht möglich ist die Zeitüberwachung für virtuelle und globale Geräte. Bei folgenden Funktionen ist die Überwachung immer ausgeblendet:</p> <ul style="list-style-type: none"><li>- bei Datensichtstationen die Eingaben (\$STEL., \$OUTIN),</li><li>- bei Platten/Disketten die Formatieraufrufe und der Peripheriespeichertransfer,</li><li>- wenn das ORG auf eine Anwenderquittung wartet.</li></ul> |
| Service-<br>Unterstützung     | <p>Die Geräteeigenschaften der Standardperipherie unterstützen den Anwender in der Sammlung von Daten aus dem Betriebsgeschehen mit der Funktion Inline-Statistik. Fehlervorbeugung und -früherkennung (Spezifizierung, Lokalisierung) sind zusätzlich mit den Funktionen Anlaufstest sowie permanenter und anstoßbarer Selbsttest möglich (Aufrufe s. 5 in /106/).</p>  |
| Generierung<br>/102/<br>/501/ | <p>Für jedes angeschlossene Gerät, mit dem über das ORG verkehrt werden soll, sind Generierangaben zwingend (logisches Gerät, Eigenschaften). Damit wird, sofern nicht generell im ORG vorhanden, implizit der entsprechende Baustein (Treiber) abgesetzt.</p>   |

---

### 5.3.1 Ein/Ausgabeeinheiten

**Geräte-  
Bausteine** Für Standardgeräte wickelt das ORG die zentrale E/A-Bearbeitung in gemeinsamen Bausteinen ab. Für die speziellen Geräteeigenschaften (s. /40/ u. folgende) von Drucker, Datensichtstation, Peripheriespeichereinheit, Datenübertragungseinheit und Magnetbandkassette stehen gerätespezifische Bausteine zur Verfügung. Der Datensichtstationsbaustein bietet für die Bearbeitung von Bedieneingaben (z.B. Ausfüllen von Bildschirmformularen) eine Sonderfunktion an. In einem Aufruf ist eine Ausgabe und eine anschließende Zeicheneingabe vereinigt (s. 21.3.13 in /101/).

**Fehler-  
reaktionen** Fehlendes ETX oder ETB führt nicht zu Fehlerreaktionen. Das Transferende ist in diesem Fall durch das Ende (die Länge) des Anwenderdatenfeldes bestimmt. Gerätefehler und Sonderzustände werden dem Anwender in den Aufrufanzeigen und über Peripherieanzeigen mitgeteilt. Das Anwenderprogramm kann also auf Sonderzustände gerätespezifisch reagieren.

Peripherieanzeigen führen in vielen Fällen zu einer Meldung auf dem Fehlermeldegerät (s. 13, bei AMBOSS 4 s. /500/). Durch entsprechende Quittung nach quittierbaren Meldungen (Handling s. 9) kann der Bediener den weiteren Ablauf beeinflussen.

### 5.3.2 Periphere Speichereinheiten

Aufrufe, E/A-Verkehr und Fehlerbehandlung wickelt das ORG standardmäßig ab (s. 5.2). Gerätefehler sind nicht quittierbar.

**Peripher-  
speicher-  
Transfer** Zwischen Magnetbandkassetten-, Magnetband- und Plattenlaufwerken ist der Datentransfer ohne Belastung des ORG möglich. Durch einen Aufruf: Peripheriespeichertransfer läßt sich beispielsweise der gesamte Inhalt eines Peripheriespeichers auf einen anderen Peripheriespeicher zeitoptimal übertragen (z.B. zur Archivierung und für Datenlieferungen; s. 21.3.19 in /101/). Der Anwender stellt einen Wechselpuffer zur Verfügung.

**Peripher-  
speicher mit  
Direktzugriff** Die Peripheriespeicher mit Direktzugriff (PSD), nämlich Plattenlaufwerke (s. /43, 44/) und Diskettenlaufwerke (s. /42/), arbeiten mit dem Gerätebaustein für PSD-Verkehr.

Der Anwender verkehrt mit PSD über die Dateiorganisation (s. 6). Bei guter Systemkenntnis ist auch Geräteverkehr möglich. Der Schreibschutz bei PSD ist mit einem Kommando oder Aufruf ein- und auszuschalten (s. 21.3.3 in /101/).

- 
- . Datenstruktur Die Aufzeichnung der Daten erfolgt in einem festsektorierten Format. Der Benutzung von Disketten und Platten geht deshalb ein Formatierungsvorgang voraus (Aufrufe s. 6 in /106/, Dienstprogramm s. /300/).
- . PSD-Cache /107/ Der PSD-Cache ist eine ORG-Funktion zur Leistungssteigerung des E/A-Verkehrs mit PSD (E/A-Aufrufe \$STEI/AU., s. 21.3.11 und 21.3.12 in /101/). Der PSD-Cache reduziert die physikalischen Zugriffe und optimiert die Transfers, indem er die benötigten Daten im HSP (in den Cachebereichen) von vornherein bereithält (Vorratshaltung, ausschnittsweise Abbilder der PSD-Daten). Der PSD-Cache organisiert den PSD-Verkehr mit den Cachebereichen in größeren und rationelleren Portionen (den Kacheln) und kann dadurch die E/A-Aufrufe weitgehend zwischen Cachebereich und Anwenderdatenfeld schneller abwickeln.

Beim ORG-Generieren ist die Cachefunktion PSD-spezifisch auszulegen. Um die Cachewirkung dem Rechnerbetrieb optimal anpassen zu können, erlauben Aufrufe und Kommandos, die Cachefunktion und ihre Modi einzustellen. Auskünfte über den Cachebetrieb lassen sich über Aufrufe oder Kommandos einholen. Für eine Standardanwendung braucht der Anwender keine oder nur wenige Angaben zu machen. Für einen dynamischen, angepaßten Cacheeinsatz wird auf die ausführliche Beschreibung der Cachefunktion und des Einsatzes in /107/ verwiesen.

Der Anwender hat außerdem die Möglichkeit, die ausgelagerte Variante des PSD-Cache zu verwenden. In diesem Fall werden die zum PSD-Cache gehörigen ORG-Bausteine im dritten Codeadrese- raum bzw. im dritten Datenadrese- raum des ORG abgelegt. Zu diesem Zweck müssen allerdings bei der Systemgenerierung die ausgelagerte Variante des PSD-Cache sowie je drei Code- und Daten- adreßräume für das ORG installiert werden.

- Magnetbandkassetten- Laufwerk Mit der Magnetbandkassette (MK) steht dem Anwender ein serieller Massen-Datenspeicher mit einem günstigen Preis-/ Leistungsverhältnis zur Verfügung, der u.a. für folgende Anwendungen geeignet ist:

- Urladen des Systems
- Datenaustausch innerhalb der Systeme
- Datensicherung (insbesondere des Systems von Festplatte)

- . Aufrufe Standard-Ein-/Ausgabe-Aufrufe sind für die Magnetbandkassetten-Laufwerke unzulässig. Es ist nur Peripheriespeicher- Transfer möglich (s.o.).

Die Bandgeschwindigkeit erlaubt hohe Datenraten, sofern der Anwender dafür sorgt, daß der "Datenstrom" in beiden Transfer- richtungen nicht abreißt (Verzicht auf Start-Stopp-Betrieb).

Mit Spezialaufrufen kann der Anwender den Datenträger bearbeiten: Folgende Bandoperationen sind u.a. möglich:

- Rückspulen auf die Bandanfangsmarke
- Umspulen bis zum physikalischen Bandende und anschließendes Rückspulen auf die Bandanfangsmarke.

- 
- . Dienstprogramm Zur Datensicherung und zum Datenaustausch mit Magnetbandkassetten steht dem Anwender das Dienstprogramm MCSAVE zur Verfügung (s. /302/).
- Magnetband-Laufwerk Das ORG stellt dem Anwender Funktionen für den Betrieb von Magnetbändern (MB, s. /46/) zur Verfügung. Bei diesem Geräteverkehr muß der Benutzer Bandbewegungen selbst vorgeben, Suchvorgänge veranlassen und die Struktur der Daten auf dem Band bestimmen; der Geräteverkehr setzt gute Systemkenntnisse voraus. Zur anwenderfreundlichen Nutzung s.u., Dienstprogramme.
- . Aufrufe
- Ein/Ausgabe-Aufrufe, s. 1.1 in /106/
  - Peripheriespeichertransfer, s. 21.3.19 in /101/
  - Magnetband-Steueroperationen (für Bandbewegungen, Aufzeichnungsverfahren, Kontrollen, Löschen, Bandmarken), s. 1 in /106/.
- . Dienstprogramme
- In der Regel verwendet der Anwender beim Einsatz von Magnetbändern die benutzerfreundlichen leistungsfähigen Dienstprogramme und -systeme
- Magnetbanddateiorganisation MBDO-M /311/
  - Magnetband- und -Dateibearbeitung TAPE-M /312/
  - Datenverwaltungssystem DVS-M /401/

### 5.3.3 Datenübertragungssteuerungen (DUST)

ANWENDER-SCHNITTSTELLE Für die Datenübertragungssteuerungen stehen dem Anwender folgende spezifische ORG-Aufrufe zur Verfügung (s. /106/):

- Daten-Ein/Ausgabe (Empfangen, Senden)
- Umladen des Partners anstoßen
- DUST sperren/freigeben
- DUST-Parametrierung
- Peripherieanzeigen erwarten
- Firmware laden

Datenpufferung Datenpufferung ist nicht zulässig, d.h. PRP, die Aufrufe für diese Geräte benutzen, sind im allgemeinen nicht räumbar.

- 
- PROGRAMM-  
TECHNISCHE  
ABWICKLUNG** Für den Anwender läuft der Vorgang der Datenübertragung in vier Schritten ab, nämlich Vorbereitung, Verbindungsaufbau, Datenübermittlung, Verbindungsabbau.
- Vorbereitung** Zum Vorbereiten des Verbindungsaufbaus bzw. der Datenübermittlung durch den Anwender gehören bei allen an der Koppelung beteiligten DUST ein definierter Anfangszustand und die Versorgung mit Parametrierdaten.
- Verbindungs-  
aufbau** Mit dem vom Anwenderprogramm abgegebenen Parametrieraufruf "Verbindungsaufbau" bekommt die DUST Sende- bzw. Empfangserlaubnis mit je nach Datenverbindung unterschiedlichen Konsequenzen.
- Daten-  
übermittlung** Nach erfolgreichem Verbindungsaufbau können die Koppelungspartner miteinander Daten austauschen. Das bedeutet, die DUST akzeptiert Sendeaufträge und meldet dem ORG empfangene Daten. Ein eventuell wartender Eingabeaufruf wird nach Datenempfang fortgesetzt.
- Alle bearbeiteten Aufträge können zu Peripherieanzeigen führen, die mit einem Aufruf abzuholen sind.
- Verbindungs-  
abbau** Der Parametrieraufruf "Verbindungsabbau" entzieht der DUST die Sende- bzw. Empfangserlaubnis. Bei Standleitungen bleibt die physikalische Verbindung bestehen, bei Wählleitungen wird damit die Verbindung automatisch abgebaut.
- FUNKTIONEN** Die schon bei der Angabe der ORG-Aufrufe angedeuteten Hauptfunktionen des ORG sollen im folgenden kurz erläutert werden.
- Parametrierung** Die veränderlichen Betriebsgrößen einer DUST lassen sich einteilen in statische Parameter (Hardware-Einstellung) und dynamische Parameter (Parametrieraufruf). Mit diesen Parametern kann der Anwender das DUST-Verhalten an vorgegebene Randbedingungen der Übertragungsstrecke anpassen sowie Zusatz- und Testfunktionen der DUST auslösen.
- Die einzelnen Aufrufparameter sind von der DUST voreingestellt. Diese Voreinstellung ist nach dem Rücksetzen der DUST bzw. nach jedem Wiederanlauf wirksam.

- 
- Standardparametrierung** Die Anwenderschnittstelle stellt drei Standardparametrierungen zur Verfügung. Der Anwender übergibt mit ihnen in zugeordneten Datenfeldern gerätespezifische Daten an die DUST.
- Grundparametrierung: Verändern der nach dem Rücksetzen der DUST wirksamen DUST-Voreinstellung (s.o.).
- Wählparametrierung: Versorgen der DUST mit den Wählziffern des Partners für die automatische Wahl und/oder Übergeben von DUST-Partneridentifikation sowie Moduseinstellung zum Überprüfen der Identifikation.
- Adreßparametrierung: Bei Mehrpunktverbindung Versorgen jedes einzelnen Kopplungspartners mit (Stations-)Adressen.
- Zusatz-/Hilfsfunktionen** Neben den o.g. Standardparametrierungen gibt es Parametrieraufrufe, die ggf. vor, während oder nach der Datenübermittlung in einer DUST spezielle Funktionen auslösen, wie
- Verbindungsaufbau
  - Verbindungsabbau
  - Urladen des eigenen Rechners anfordern
  - Urladevorgang im Partnerrechner anstoßen
- Peripherieanzeigen erwarten und abholen** Alle DUST können in jeder Betriebsphase Peripherieanzeigen liefern, die der Anwender mit einem Aufruf abzuholen hat. Die Bedeutung der Anzeigenbits ist gerätespezifisch.
- Die Anzeigen-Ursachen sind zwei Klassen zuzuordnen, Anzeigen aufgrund von Fehlern (Übertragungsfehler, Hardwarefehler oder Programmierfehler) und Anzeigen aufgrund von Ereignissen (interne Zustandswechsel, Verbindungsabbau, u.a.).
- Daten senden/empfangen** Das ordnungsgemäße Abwickeln von Sende- und Empfangsaufträgen setzt voraus, daß der Anwender alle erforderlichen Software- und Hardware-Parametrierungen DUST-spezifisch und sinnvoll vorgenommen hat.
- Mit dem Ausgabeaufruf initiiert der Anwender Sendeaufträge an die DUST und spezifiziert über Parameter DUST-abhängig Transferdetails.
- Ein vom Anwender abgegebener Eingabeaufruf wird erst wirksam, wenn die DUST dem ORG den erfolgten Datenempfang durch einen Anruf mitgeteilt hat. Außer den Daten, die in das bereitgestellte Datenfeld gelangen, erhält der Anwender DUST-abhängig Zusatzinformationen in den Parameterblock des Eingabeaufrufs eingetragen.
- Urladen** In homogenen Rechnernetzen der Modellreihe SICOMP | 6.680 kann der Anwender die DUST auch zum Urladen von Partnerrechnern über Rechnerkopplungsstrecken benutzen (dabei Urladeformat der Zentraleinheiten beachten! s. /25/, /26/).



---

Urladen ist in jeder Betriebsphase der DUST erlaubt. Bei Punkt-zu-Punkt-Verbindung ist Urladen in beiden Transfer-richtungen zulässig. Bei Mehrpunktverbindung ist nur Urladen von Trabantenstationen durch eine Leitstation möglich.

**DUST sperren/  
freigeben** Aufgrund eines Sperr-Aufrufs setzt das ORG die angesprochene DUST zurück und beendet jeden noch für dieses Gerät anstehenden Aufruf mit Aufrufanzeige. Aufrufe an ein gesperrtes Gerät werden vom ORG ebenfalls mit Aufrufanzeige abgewiesen. Die Freigabe erfolgt mit Freigabe-Aufruf.

**Testbetrieb** Zum Erleichtern der Inbetriebnahme und zum Test einer Kopplungsstrecke bieten die DUST unterschiedliche Tests und Hilfsmittel.

Spiegeltest: Die DUST sendet ein vom Anwender bereitgestelltes Datenfeld zur Partner-DUST und übergibt dem Anwender die vom Partner unverändert zurückgesandten (gespiegelten) Daten nach formaler Prüfung.

Durchreichtest: Die DUST übergibt dem Anwender jedes empfangene und auf formale Richtigkeit überprüfte Anwender-Datenfeld unverändert. Die entsprechenden Ein- und Ausgaben sind auf beiden Seiten der Kopplungsstrecke mit Aufrufen vorzunehmen.

Die sinnvolle Benutzung beider Tests gestattet dem Anwender, die einwandfreie Datenübermittlung einer Kopplungsstrecke zu prüfen. Mit einem Aufruf sind beide Tests ausschaltbar.

**STANDARD-KOPP-  
LUNGSSOFTWARE** Für Kommunikationsaufgaben in vermaschten Netzen - auch mit systemfremden Rechnern - stehen dem Anwender Subsysteme zur Verfügung (s. /403/).

#### 5.4 Datenpufferung

**Überblick** Die Datenpufferung ist eine bei der Systemgenerierung wählbare Funktion des ORG. Sie übernimmt Standard-E/A-Aufrufe der Programme an langsame Geräte, puffert die Aufrufdaten auf PSD bzw. im HSP und veranlaßt - stellvertretend für die aufrufgebenden Programme - die Operationen nach Maßgabe der Geräteverfügbarkeit.

Damit lassen sich kurzfristige hohe E/A-Anforderungen der Programme an die Arbeitsgeschwindigkeit der Geräte anpassen. Als Hauptvorteil ergibt sich eine Reduktion der Verweilzeiten der PRP in den Laufbereichen mit entsprechend verbesserter Anlagenausnutzung. Die Bearbeitung von Aufrufen aus HRP ist in gleicher Weise möglich, soweit sinnvoll und vertretbar.

- 
- Varianten** Die Datenpufferung steht in zwei Ausführungen, dem "schnellen" (S) und dem "langen" (L) Modus zur Verfügung.
- S-Modus** Die Pufferung im schnellen Modus bearbeitet nur Ausgabeaufrufe. Das Programm (HRP oder PRP) wird nach Übernahme der zu puffernden Daten auf PSD stets fortgesetzt, ohne Aufrufausführungsanzeigen zu erhalten. Optionell kann im Fehlerfall (Originalgerät unklar) auf ein Ersatzgerät umgeschaltet werden. Die der Pufferung unterstehenden Geräte lassen sich mit SBP-Kommandos als "klar" oder "unklar" erklären. Eine ähnliche Funktion bietet das System SPOOL-M (s. /402/).
- L-Modus** Im langen Modus lassen sich E/A-Aufrufe im HSP puffern. Das Anwenderprogramm (nur PRP) wird - abhängig von den zugeordneten Aufrufen - fortgesetzt oder wartend, ist aber räumbar. Die Datenpufferung bewirkt also, daß der Anstoß eines E/A-Verkehrs kein Sperrereignis für PRP darstellt (Programme räumbar, s. 4). Das Anwenderprogramm erhält nach Operationsausführung über das Operationsergebnis die üblichen Anzeigen und die Adresse des letzten übertragenen Zeichens.
- Abgrenzung** Für Eingabeaufrufe ist nur der lange Modus sinnvoll und zulässig.
- Die Pufferung im schnellen Modus lohnt in der Regel nur für Geräte und Aufrufe, bei denen die Zeit für eine Aufrufpufferung (PSD-Transfers) kürzer als Direktausführung ist. Der lange Modus kann auch verwendet werden, um zusätzliche Räumbarkeitsstellen zu schaffen. Da der lange Modus eine Pufferung nur im HSP durchführt, tritt bei normalem Ablauf (ohne Räumung) keine merkliche Aufrufverzögerung auf.
- Besonderheit** Jedem der schnellen Datenpufferung unterstellten (Ausgabe-) Gerät kann ein Ersatzgerät zugeordnet werden. Bei unklarem "Original-"Gerät wird auf das Ersatzgerät umgeschaltet, d.h. der Aufruf wird für dieses noch einmal vollständig ausgegeben.
- Anwenderschnittstelle** Alle Standard-Transferaufrufe an die vorgenannten Geräte laufen (sofern generiert) über die Datenpufferung ab. Der Anwender kann jedoch über den GEDA-Block Ausnahmen erreichen. Dort entscheidet er auch über den Modus und ob E/A-Aufrufe ohne Laufbereichssperrung arbeiten.
- Koordinierung** Nicht von der Datenpufferung zu übernehmende Aufrufe gelangen unmittelbar in die Transfer-Warteschlange des betroffenen Gerätes. Sie können deshalb von der Datenpufferung zu bearbeitende ältere Aufrufe überholen. Das gleiche gilt für Aufrufe aus HRP.
- Bei einer derartigen Mischung von Aufrufen mit schneller, langer und ohne Datenpufferung muß der Benutzer etwaige Koordinierungsfolgen selbst bedenken.

Pufferung im S-Modus Je Aufruf werden die Daten, der Aufruf-Parameterblock und organisatorische Informationen in einem HSP-Universalpuffer aufbereitet und in einem Pufferelement auf dem PSD hinterlegt. Die Menge aller Pufferelemente für ein Gerät ist eine Pufferzone. Die Pufferzonen für alle betroffenen Geräte bilden eine Pufferzonendatei auf einem PSD, die der ORG-Generator nach Anwenderangaben einrichtet.

Treffen Aufrufe schneller ein als sie bearbeitet werden können (z.B. auch bei unklarem Gerät), so kann die Pufferzone überlaufen (kein freies Pufferelement verfügbar). Der schnelle Modus bietet für diesen Fall zwei Bearbeitungsarten - Anzeigen- oder Warteschlangenbildung - die der Anwender bei der Generierung gerätespezifisch festlegt. Mit Aufrufen kann der Anwender erreichen, daß gepufferte Ausgabeaufrufe unterdrückt werden bzw. neu eintreffende funktionslos bleiben, bis die ursprünglichen Eigenschaften der Datenpufferung wieder aktiviert werden sollen.

Pufferung im L-Modus Jedem Gerät wird ein HSP-Puffer zugeordnet, den es u.U. mit anderen Geräten teilen muß. Jeder Puffer wird über eine eigene Warteschlange koordiniert. Ist der Puffer unbelegt (WS leer), wenn ein E/A-Aufruf abgegeben wird, überträgt das ORG sofort das Datenfeld mit den erforderlichen Aufrufparametern in den Puffer und setzt dann das PRP fort, wenn kein implizites Warten gewünscht. Ist der Puffer belegt, wird der Aufruf lediglich in die Puffer-WS eingetragen und das PRP fortgesetzt bzw. wartend wie bei leerer WS. Bei der Weitschaltung der WS werden die Daten des nächsten Aufrufs in den Puffer gebracht.

### 5.5 Verkehr mit Prozeßperipherie

Hardware Den Verkehr mit der Prozeßperipherie kann der Anwender über die Prozeßeinheiten (PE) 3600 und/oder F7 abwickeln. Der Austausch der Prozeßinformationen ist über folgende Prozeßsignalformer (PSF) möglich:

| Signalformer             | PE 3600 | PE F7 |
|--------------------------|---------|-------|
| Digitaleingaben          | x       | x     |
| Digitalausgaben          | x       | x     |
| Analogeingaben           | x       | x     |
| Analogausgaben           | x       | x     |
| Digital-Ein/Ausgaben     | x       | -     |
| Zähler                   | -       | x     |
| Zeitgeber                | x       | -     |
| Digital-E/A(220V-Signal) | -       | x     |

Aufrufe Da bei Geräten der Prozeßperipherie Aufruf-Standardisierung und Austauschbarkeit unvereinbar sind, stehen für die häufigsten Anwendungen geräte- bzw. funktionsspezifische Makroaufrufe zur Verfügung. Sie führen zu im Anwenderprogramm ablaufenden Befehlsfolgen und ORG-Aufrufen.

Funktionen Das ORG bietet Funktionen für "schnelle" Prozeßsignalformer, zur Alarmbearbeitung und für die integrierende Analogeingabe.

---

### 5.5.1 Funktionen für "schnelle" Prozeßsignalformer

Bei der Gruppe der sogenannten schnellen Prozeßsignalformer (PSF) erfolgt der Verkehr zwischen Zentraleinheit und PSF über E/A-Befehle mit zentraler Initiative.

- Aufrufarten** Es stehen elementare Makroaufrufe für den Transfer einzelner Digital- und Analogwerte (kurz: Elementaraufrufe) und listenorientierte Makroaufrufe für den Transfer mehrerer Datenwerte gemäß einer Liste (kurz: Listenaufrufe) zur Verfügung.
- Elementaraufrufe** Die Aufrufe setzen direkt Befehlsgruppen aus E/A-Maschinenbefehlen im Anwenderprogramm ab, die ohne das ORG ablaufen. Die Aufrufe stehen als registerorientierte Aufrufe (Parameter stehen in Registern) oder als speicherorientierte Aufrufe (Parameter in Hauptspeicherzellen) für Digital-E/A und Analogausgabe zur Verfügung. Parameter sind: EA-Adresse der Prozeßeinheit, PSF-Adresse und Meßwert.
- Listenaufrufe** Es laufen wie bei Elementaraufrufen nur Befehle des aufrufgebenden Anwenderprogramms ab. Die Aufrufe bestehen aus Aufrufanstoß, Parameterblock und GEDA-Block und dienen dem Verkehr mit Meßwertlisten.

### 5.5.2 Funktionen zur Alarmbearbeitung

**Bearbeitungsarten** Für die Alarmbearbeitung stehen dem Anwender im ORG verschiedene Möglichkeiten zur Verfügung, die sich in der Reaktions-schnelligkeit und im Anwendungskomfort unterscheiden:

- Zeitoptimale Alarmbearbeitung
- Bearbeitung peripherer Organisationsanforderungen
- Alarmgesteuerte Koordinierung

**Zeitoptimale Alarmbearbeitung** Diese Funktionen dienen Aufgaben der Alarmbearbeitung, bei denen die Reaktionszeit zwar eine überragende Rolle spielt, andererseits jedoch auf Anwendungskomfort in angemessenem Umfang nicht verzichtet werden kann.

Alarmbearbeitende Programme melden sich mit Anmeldeaufrufen beim ORG als Empfänger für bestimmte periphere Unterbrechungsereignisse an, die damit auch freigegeben werden (Signalformerfreigabe), und warten anschließend mit einer zugeordneten Wartefunktion auf das Eintreffen solcher Ereignisse (Interrupt). Die Anmeldung gilt so lange, bis sie explizit durch einen auf sie bezugnehmenden Abmelde-Aufruf widerrufen wird. Die Gültigkeit einer Anmeldung wird also durch eintreffende Unterbrechungsereignisse nicht beeinträchtigt.

Da nur das Warten, nicht aber die Anmeldung nach jeder Unterbrechung erneuert wird, ergeben sich kurze Reaktionszeiten. Im Anmelde-Aufruf sind die Prozeßsignalformer anzugeben, deren Alarme das Programm bearbeiten soll.

Wird das alarmverarbeitende Anwenderprogramm wegen eingetretener Alarme vom ORG fortgesetzt, so muß der Anwender die Anzeigen des Anmelde-Aufrufs abfragen sowie am "Alarmbit" aller angemeldeten Prozeßsignalformer im PB feststellen, für welche PSF der Alarm gilt. Von diesen PSF sind die Alarmwörter einzulesen.

Anmelde-Aufruf, Abmelde-Aufruf und der allgemeingültige Warte-Aufruf sind datendefinierende Parameterblöcke. Ihre Funktionen werden durch den Anstoß-Aufruf an das ORG ausgelöst. Die Aufruf-Parameterblöcke verweisen auf einen GEDA-Block ohne Dateierklärung, wie er für Ein-Ausgabe-Aufrufe allgemeingültig festgelegt ist. Er bezieht sich auf das logische Gerät mit eigener Geräteerkennung, das alle PSF der Prozeßeinheit umfaßt.

**Bearbeitung peripherer Anforderungen** Eine einfache Schnittstelle gestattet dem Anwender, durch Aufrufe und E/A-Maschinenbefehle in der Anwenderebene Prozeßeinheiten ohne das ORG zu betreiben, die periphere Anforderungen bzw. Interrupts abgeben.

**Alarmgesteuerte Koordinierung** Eine Erweiterung der Funktionen zur zeitoptimalen Alarmverarbeitung gestattet - nicht nur dem PEARL-Anwender (s. /208/) - mit dem Eintreffen von Prozeßereignissen (Alarmen) bestimmte Koordinierungsfunktionen zu aktivieren.

Mit einem Anmeldeaufruf teilt der Anwender dem ORG die Koordinierungsfunktion und die sie auslösenden Prozeßsignalformer mit. Nach jedem Eintreffen eines Alarms wird die zugehörige Koordinierungsfunktion ORG-intern angestoßen, das zugehörige Alarmwort gelesen (PSF-Freigabe) und nach einer ODER-Verknüpfung mit dem korrespondierenden alten Alarmwort abgelegt.

Durch einen komplementären Aufruf, der nicht vom "anmeldenden" Programm abgegeben werden muß, werden alle im ORG registrierten alarmgesteuerten Koordinierungsaufträge gelöscht, die eine bei der Anmeldung angegebene Vergleichskennung tragen.

### 5.5.3 Integrierende Analogeingabe

**Betriebsart** Der Verkehr mit der integrierenden Analogeingabe (IAE) läuft hardwarebedingt wie folgt ab:

- Anstoß der Meßwertintegration mit PSF- und Meßstellenanwahl sowie Versorgung mit Meßvorschrift.
- Eintreffen einer Unterbrechungsanforderung als Abschlußmeldung für die beendete Meßwertintegration.
- Einlesen des integrierten Meßwertes in eine HSP-Zelle.

Für den Verkehr mit der IAE stehen Makroaufrufe und Ein-/Ausgabe-Maschinenbefehle zur Verfügung. Der Anwender bestimmt den Grad der Zusammenarbeit mit dem ORG und die übrige programmtechnische Abwicklung durch die Wahl der Aufrufvariante (Variante 1 und 2) und die Art der verwendeten Meßstellenlisten.

## 5.6 Anwendertreiber

|                                     |  |
|-------------------------------------|--|
| Definition                          | <p>Ein Anwendertreiber ist ein vom Benutzer implementiertes Programm, das für anwenderspezifische periphere Einheiten (Sonderperipherie) die Funktionen eines Gerätetreibers übernimmt. Diese Geräte sind nach Installation des Anwendertreibers wie folgt ansprechbar:</p> <ul style="list-style-type: none"><li>- über das ORG mit Standard-E/A-Aufrufen (Online-Betrieb, s.u.) oder</li><li>- direkt mit Maschinen-Befehlen und Koordinierungsaufrufen (Offline-Betrieb, s.u.) auch geeignet zur schnellen Reaktion auf Alarme.</li></ul> |
| Zulässige Geräte                    | <p>Im Offline-Betrieb ist jede Sonderperipherie zulässig, die der physikalischen E/A-Schnittstelle der Modellreihe genügt (s. /105/).</p>  |
| Installation                        | <p>Diese Geräte sind mit einer vom System vorgegebenen Geräteerkennung zu generieren. Der Anwendertreiber ist wie ein Programm als HRP durch Laden in das System einzubringen.</p>   |
| Zuordnung<br>Gerät-Treiber          | <p>Mit der Geräte-Treiber-Zuordnung des Anwenders schließt das ORG das anwenderspezifische Gerät an den Anwendertreiber an. Diese Zuordnung ist durch Aufruf oder SBP-Kommando vorzunehmen und aufzuheben. Der Anwendertreiber läuft höherprior als das ORG ab.</p>  |
| Anwender-<br>treiber-<br>Funktionen | <p>Der Anwendertreiber hat die im Gerätebaustein des ORG nicht realisierte Funktion des gerätespezifischen Treibers, wie gerätespezifische Operationen, Geräteanstoß und Bearbeitung des Geräteinterrupts, wahrzunehmen.</p>   |
| Online-<br>Betrieb                  | <p>Standard-E/A- und Service-Aufrufe für anwenderspezifische Geräte wickelt das ORG (wie Aufrufe für Standardgeräte) über zentrale E/A-Bearbeitung und Gerätebaustein (für anwenderspezifische Geräte) ab. Das ORG aktiviert den Anwendertreiber nur für den Geräteanstoß und den Interrupt.</p>   |
| Offline-<br>Betrieb                 | <p>Der Anwender erwartet vom anwenderspezifischen Gerät einen Alarm oder Anruf, mit dessen Eintreffen der Anwendertreiber zunächst die Ein-/Ausgabe abwickelt, danach das wartende Programm fortsetzt.</p>   |

---

## 6 Dateiorganisation

Dateien gemäß 1.2 und 2.3 auf Peripherspeichern mit wahlfreiem Zugriff (s. 5.3.2) werden von der Siemens-Dateiorganisation verwaltet.

**Anwenderschnittstelle** Der Anwender verkehrt mit Dateien über ORG-Aufrufe (dateiorganisatorische Funktionen und Transferaufrufe). Die Parameter stehen in GEDA-Blöcken.

Die Dateiorganisation ist für Platten und Disketten kompatibel (gleiche Buchführung, gleiche Aufruffunktionen). Beim Einsatz von Disketten ist der langsamere Datenverkehr (ca. Faktor 10) und die geringere Speicherkapazität gegenüber Platten zu berücksichtigen.

**Generieren** Das Gerät ist beim Generieren als Randomspeicher im Geräte-Parametersatz zu definieren. Die Bausteine der Dateiorganisation sind ohne Generierwunsch segmentiert generiert (geringer HSP-Platzbedarf). Mit Generierwunsch setzt der ORG-Generator alle Bausteine der Dateiorganisation im HSP ab. Beim Generieren hat der Anwender auch die Länge des Arbeitspuffers festzulegen.

**Ablauf** Das zeitaufwendige Durchsuchen der Namens- und Platzbuchführung läuft in der Anwender-Ebene ab, in der der Aufruf erfolgte.

**Datenträger, Geräte** Datenträger für Dateien sind Fest- und Wechselplatten und Disketten. Die Dateiorganisation kennt nur logische Datenträger und Geräte. Ihre physikalische Einheit (gemeinsamer Plattenstapel, Laufwerksachse) tritt in der Regel nicht auf. Auf den Datenträgern wird Platz immer in ganzen Sektoren (512 byte) vergeben.

**Datenträger-vorbereitung** Die Dateiorganisation setzt eingerichtete Datenträger voraus, d.h. eine für den Dateizugriff vorbereitete Buchführung auf dem Datenträger. Das Einrichten erfolgt mit dem Programm INITM und entscheidet auch über die Dateianzahl (s. /300/).

Datenträgerwechsel s. 6.3

**Datenträgerblock** Er steht innerhalb des ORG und enthält Informationen über den im Gerät eingelegten Datenträger (als Erweiterung des Geräteblocks, Geräteliste).

**Datenträgeridentifikation** Sie erfolgt durch einen beim Einrichten des Datenträgers zwingend anzugebenden Datenträgernamen, der auf dem Datenträger geführt und vom ORG beim Datenträgerwechsel (s. 6.3) und bei den dateiorganisatorischen Funktionen (s. 6.1) ausgewertet wird.

---

|                                      |   |
|--------------------------------------|---|
| Datei-<br>buchführung                | Die Buchführung zu den Dateien eines Datenträgers wird auf dem Datenträger (Platz durch INITM vorgegeben), in Teilabbildern davon im ORG-Listebereich im HSP, in ORG-Transfer-Warteschlangen und in den GEDA-Blöcken und Hilfsfeldern in den Anwenderprogrammen geführt.  |
| Zugriff zur<br>Datei-<br>buchführung | Anwenderprogramme können mit Hilfe von Aufrufen an das ORG das Etikett einer Datei und die Dateibuchführung lesen sowie das Etikett neu schreiben (s. 2.3).   |
| Dateiarten                           | Als Besonderheit bei Dateien sind lediglich Prozeßdateien durch eine entsprechende Kennung im GEDA-Block beim Einrichten angebar. Ihre Buchführung bleibt vom Einrichten bis zum Löschen im HSP, auch über Wiederanlauf hinweg. Das ORG löscht keine Dateien nach Wiederanlauf.   |
| Datei-<br>identifikation             | Sie erfolgt durch einen vom Anwender beim Einrichten der Datei festzulegenden Dateinamen in Verbindung mit dem logischen Gerätenamen. Das ORG legt keine Dateinamen fest. Namen bestehen aus sechs Zeichen und müssen je Gerät eindeutig sein.<br><br>Ein Punkt als viertes Zeichen kennzeichnet im System Bibliotheken, unterbindet die Dateinamensprüfung durch das ORG und ist deshalb für Anwenderdateien unzulässig. |
| Dateianzahl                          | Je logisches Gerät sind bei Platten 7808 Dateien, bei Disketten 976 Dateien möglich. Die für ein logisches Gerät max. mögliche Dateianzahl wird beim Einrichten des Datenträgers angegeben.   |
| Dateilänge<br>und -lage              | Die Dateilänge (minimal 1 byte, maximal 65536 Sektoren) ist beim Einrichten vorzugeben, aber mittels ORG-Aufruf änderbar. Unter bestimmten Voraussetzungen ist Dateistückelung möglich. Dateien beginnen am Sektoranfang. Der Dateianfang ist an beliebiger Stelle vorgebar (mittels PSD-Sektornummer) oder wird von der Platzverwaltung ermittelt.   |
| Dateizeiger                          | Er wird vom ORG bei serielltem Zugriff als relative Byteadresse im GEDA-Block des Anwenders geführt und bezeichnet die Stelle, ab der der nächste serielle Transfer erfolgt. Mit Aufrufen läßt sich der Zeiger einstellen und lesen.  |
| Datensätze                           | Ein logischer Satz ist die Menge der durch einen Aufruf übertragenen Daten und ist<br><ul style="list-style-type: none"><li>- bei alphanumerischen Aufrufen durch das Endezeichen (ETX/ETB) begrenzt oder durch die Satzlängenangabe im GEDA-Block bestimmt (die zuerst erreichte Grenze gilt) und</li><li>- bei binären Aufrufen nur durch die Satzlängenangabe im GEDA-Block bestimmt.</li></ul>                        |



---

Satzlänge Null bedeutet keine Satzlängenangabe und hat die Wirkung einer übergroßen Angabe. Unterschiedliche Satzlängen in der Datei muß der Anwender selbst verwalten.

Datenarten      Daten in einer Datei können durch zugeordnete Aufrufe binär oder alphanumerisch zu interpretieren sein (s. auch 5). Binäre Operationen sind bei alphanumerischen Daten zulässig, jedoch unterbleibt die Erkennung von Steuerzeichen.

Zugriffsarten    Es sind serieller und direkter Zugriff möglich.

    seriell      Hier greift der Anwender mit Aufrufen für seriellen Zugriff zu den Daten beliebiger Datenart in der Reihenfolge ihrer Anordnung innerhalb der Datei zu. Der Dateizeiger wird beim Eröffnen auf  $\emptyset$  gesetzt und durch Transfers um Satz- oder Puffergrößen weitergeschaltet.

    direkt      Hier nennt der Anwender in den Aufrufen für direkten Zugriff eine Byteadresse innerhalb der Datei, ab der übertragen wird. Der Dateizeiger wird nicht verstellt.

Koordinierung    Für den Dateiverkehr sind alle Koordinierungsleistungen des ORG wie bei der Ein-Ausgabe verfügbar. Das sind Wartefunktionen (s. 4.3) und der Koordinierungszähler (s. 4.4). Die Transferkoordinierung erfolgt mit einer Warteschlange je Laufwerk. Koordinierung der dateiorganisatorischen Aufrufe s. 6.1.

Schutz-  
funktionen      Möglich sind Schreibschutz und Dateischutz.

.Schreibschutz    Schreibschutz verhindert Schreiben in eine Datei durch Zurückweisung des schreibenden Aufrufs. Der Schutz ist durch Etikettänderung unter Verwendung der Etikettaufrufe einzuschalten.

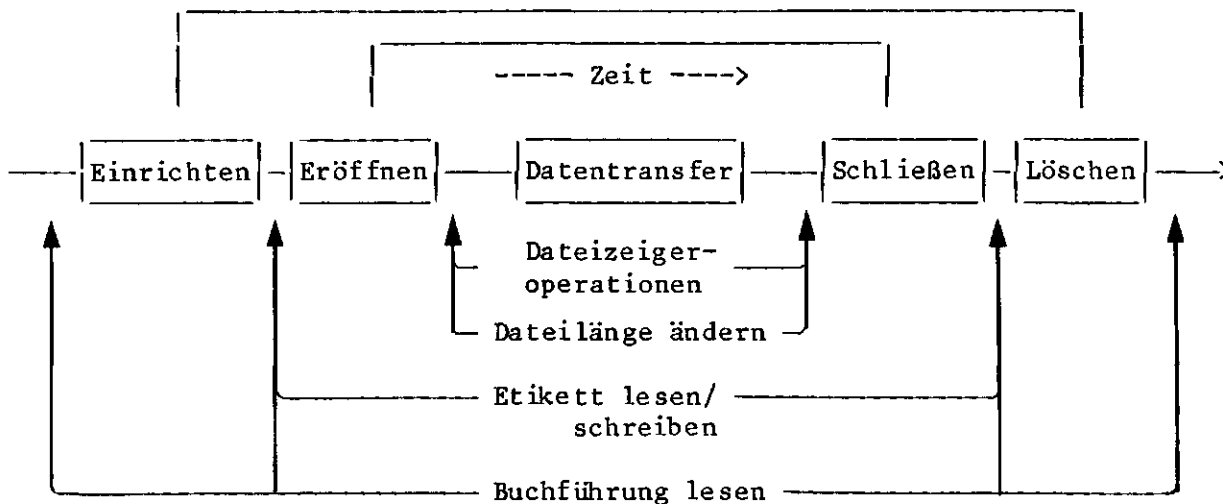
.Dateischutz      Dateischutz ist realisierbar durch ein Benutzerkennzeichen und ein Eigentümerkennzeichen. Beide lassen sich beim Einrichten angeben oder durch Aufruf ändern und durch einen Parameter im GEDA-Block als wirksam erklären. Sie müssen beim Zugriff zur Datei genannt sein.

Fehlendes Benutzerkennzeichen im GEDA-Block bewirkt Abweisen eines Eröffnungsaufwurfes. Fehlendes Eigentümerkennzeichen im GEDA-Block-Hilfsfeld bewirkt Abweisen der Aufrufe Länge ändern, Löschen, Etikett schreiben.

### 6.1 Dateiorganisatorische Funktionen

Dem eigentlichen Dateiverkehr mit Transferoperationen sind organisatorische Funktionen wie folgt zugeordnet. Sie werden, wie Transfers, durch Aufrufe veranlaßt. Zu allen Aufruf-Parameterblöcken gehört ein GEDA-Block, der den Hauptteil der Parameter enthält. Bei bestimmten Aufrufleistungen muß der Anwender ein GEDA-Block-Hilfsfeld zur Verfügung stellen.

Für das Arbeiten mit einer Datei ergibt sich damit folgendes Schema:



**Einrichten, Löschen** Damit eine Datei im System registriert und gebrauchszugänglich wird, ist sie vom Anwender durch ein beliebiges Programm einzurichten. Eine nicht mehr benötigte Datei läßt sich durch Löschen aus der Buchführung entfernen. Für beide Funktionen stehen auch Kommandos zur Verfügung.

**Eröffnen, Schließen** Ein Programm muß den Verkehr mit einer Datei durch ihr Eröffnen einleiten. Dabei wird der Dateizeiger eingerichtet und Buchführungsinformation ausgetauscht. Die Umkehrfunktion zum Eröffnen ist das Schließen, das Buchführungselement wird freigegeben.

**Dateizeigeroperationen** Mit dem Lesen und Einstellen des Zeigers lassen sich Bearbeitungsschritte modifizieren.

**Dateilänge ändern** Reicht die beim Einrichten angegebene Dateilänge später nicht aus, so ist eine Verlängerung möglich.

**Etikett lesen, schreiben** Diese Funktionen dienen zum Prüfen und Ändern von Dateiparametern.

**Buchführung lesen** Mit dieser Funktion lassen sich das Datenträgeretikett, die Speicherbelegungstafel oder das Dateietikett lesen.

**implizite Ausführung** Von den obigen Funktionen lassen sich implizit in Verbindung mit dem ersten Transfer oder einer Dateizeigeroperation durch zugeordnete GEDA-Block-Kennung ausführen: Eröffnen bzw. Einrichten und Eröffnen.

**Prüfungen** Das ORG prüft weitgehend in dem aus Anzeigen und Meldungen ersichtlichen Umfang die Aufrufe, ihre Parameter sowie GEDA-Block und Hilfsfeld auf Richtigkeit und Vollständigkeit. Benötigt die Aufruffunktion ein GEDA-Block-Hilfsfeld, so prüft das ORG, ob die angegebene Adresse in einer geladenen Seite liegt, andernfalls erfolgt Fehlermeldung.

Im Dateiverkehr kann der Anwender als zusätzliche Sicherheit bei der Identifikation neben dem logischen Gerätenamen und dem Dateinamen auch den Datenträgernamen im GEDA-Block nennen (s. Datenträgeridentifikation). Fehlt eine solche GEDA-Block-Angabe, so arbeitet das ORG ohne Anzeige oder Meldung weiter.

**Fehlerfälle** In Fehlerfällen bei dateiorganisatorischen Aufrufen bricht das ORG den Aufruf ab, stellt den Zustand wie vor dem Aufruf her und setzt das Programm fort. Bei Fehlern in kombinierten Aufrufen (die in Einzelaufrufe zerlegt werden) wird der vor dem fehlerhaften Einzelaufruf aktuelle Zustand hergestellt.

**Koordinierung** Das ORG koordiniert alle dateiorganisatorischen Aufrufe aller angeschlossenen PSD über eine Warteschlange. Deshalb kann es zu Verzögerungen bei diesen Aufrufen kommen. Die Programme warten an der Aufrufstelle, PRP sind nicht räumbar.

**Zeit-optimierung** Drei Möglichkeiten erlauben, auf unterschiedliche Weise den Dateiverkehr zu optimieren:

- Hauptspeicherresidente Dateiorganisation (Generierwunsch). Es werden die Segmentwechsel eingespart.
- optimale Arbeitspufferlänge. Sparen von Zugriffen zur PSD-residenten Dateibuchführung.
- Der Verkehr mit Prozeß- oder eröffneten Dateien erfordert keine Zugriffe zur externen Buchführung.

## 6.2 Datei-Ein-/Ausgabe, PSD

**Transfer-aufrufe** Die Aufrufe sind abhängig von Daten- und Zugriffsart zu wählen:

| <u>Richtung</u>                          | <u>Zugriff</u> | <u>Datenart</u>                          | <u>Aufruf</u>   |
|--|----------------|--|---|
| Lesen<br>Schreiben<br>Lesen<br>Schreiben | } seriell      | binär<br>binär<br>alphanum.<br>alphanum. | } Standard-E/A für<br>Dateiverkehr<br>mit seriellem<br>Zugriff          |
| Lesen<br>Schreiben<br>Lesen<br>Schreiben | } direkt       | binär<br>binär<br>alphanum.<br>alphanum. | } Standard-E/A für<br>Datei- u. Geräte-<br>verkehr mit<br>Direktzugriff |

**Anwender-angaben** Die Datei muß - ggf. implizit - eröffnet, Transferrichtung und Schreibschutz müssen verträglich sein. Der Anwender hat darüber hinaus die größte mögliche Transferlänge, Datenfeldlänge und -länge sowie den gewünschten Zugriff anzugeben.

---

ORG-Leistungen Dem Transfer gehen ggf. implizite dateiorganisatorische Funktionen voraus. Das ORG transferiert die Daten zwischen Datenfeld und Datei und hinterlegt bei Aufrufabschluß im Parameterblock die um 1 erhöhte Adresse des letzten transferierten Datums als Byteadresse. Bei serielltem Zugriff erhöht es außerdem den Dateizeiger um die Länge des transferierten Feldes.

### 6.3 Datenträgerwechsel im Betrieb, PSD

Wechseldatenträger, die keine Systemdateien enthalten, können unter Beachtung einiger Bedingungen ohne Betriebsunterbrechung ausgetauscht werden.

ORG-Leistungen Das ORG bietet Hilfen im Rahmen der Dateiorganisation an (Kassettenwechselbaustein, kein besonderer Generierwunsch). Das ORG erkennt Aus- und Wiedereinschalten von PSD-Laufwerken aufgrund von Peripherieanzeigen beim ersten Zugriff. Es prüft dabei die Datenträgerbuchführung, indem es den alten Datenträgernamen mit dem neuen vergleicht, soweit die Geräte der Dateiorganisation unterstehen.

Voraussetzungen Ein unvorschriftsmäßiger Datenträgerwechsel kann Zerstörung der PSD-Daten, des ORG oder des Anwendersystems zur Folge haben oder einen Wiederanlauf bedingen.

Platten oder Disketten mit unterschiedlichen Dateninhalten müssen unterschiedliche Datenträgernamen haben, sonst ist ein Wechsel vom ORG nicht erkennbar. Zur Festlegung der Namen s. die Programme INITM und COPY-M (s. /300/, /301/).

Laufwerke, die den Abbildspeicher oder den Wartebereichs-  
speicher enthalten, dürfen nicht abgeschaltet werden.

Verboten ist der Wechsel von Systemdatenträgern, auf denen beim Systemgenerieren Systemdateien eingerichtet wurden. Segmentspeicherwechsel ist möglich, wenn noch ein weiterer Segmentspeicher zur Verfügung steht (s. 1).

Ein Laufwerk/Gerät darf nur abgeschaltet (unklar) werden, wenn in diesem Zeitraum kein Ein-/Ausgabeverkehr mit diesem Laufwerk stattfindet (auch kein ORG-interner). Vorsicht beim Wechsel, wenn die Festplatte desselben Laufwerks Systemdateien enthält!

Ablauf Ein Datenträgerwechsel ist nur mit Off- und Online-Schalten des Laufwerks zulässig.

---

## 7 Simulation

|                       |   |
|-----------------------|---|
| Arten                 | Es gibt System-Simulationen und Anwender-Simulationen.  |
| System-Simulationen   | System-Simulationen sind Erweiterungen des Befehlsvorrats, Konvertierungs- und Hilfsroutinen. Sie sind Bestandteile des ORG-Masterstapels, die bis auf Arithmetiksimulationen und Code-Umwandlung immer abgesetzt werden. Die Simulation der Rechnungstypen (D, G und K), die Code-Konvertierung (EBCDIC.S) sowie die Simulation ZEO3 (d.h. Simulation von ZEO3-Befehlen auf ZEO1) kann der Anwender durch Generierwünsche auswählen.   |
| Anwender-Simulationen | Sie sind bei Bedarf vom Anwender nach bestimmten Konventionen zu erstellen und bei der ORG-Generierung in das System einzubringen.  |
| Konventionen          | Simulationsroutinen arbeiten mit einem eigenen Registersatz und haben Zugang zum Standardregistersatz. In Simulationsroutinen sind nicht zugelassen: <ul style="list-style-type: none"><li>- ORG-Aufrufe</li><li>- Datenübergabe in Registern</li><li>- Hilfszellen</li><li>- Simulationen (d.h. keine Schachtelung).</li></ul>   |
| Ansprung              | System-Simulationen werden automatisch (ORG-intern) veranlaßt oder durch Hardware-Befehle, die als Datendefinitionen zu schreiben sind, in Anwenderprogrammen angesprungen. Daneben gibt es Makroaufrufe.<br><br>Anwender-Simulationen muß der Anwender ebenfalls mit einem Hardware-Befehl ansprechen, soweit er sich nicht selbst Makroaufrufe schafft.   |
| Verlassen             | Jede Simulation wird mit einem Hardware-Befehl verlassen.   |
| Simulation ZEO3       | Der gegenüber der ZEO1 erweiterte Befehlsvorrat der ZEO2, ZEO3 und ZEO4 (s. /25/, /26/) macht auf der ZEO1 Simulationen für folgende ZEO3-Befehle erforderlich: <ul style="list-style-type: none"><li>- Bytefeld transferieren</li><li>- Bytefeld vergleichen</li><li>- Byte tauschen</li><li>- Byte vergleichen</li><li>- ungleiches Zeichen suchen</li><li>- gleiches Zeichen suchen</li><li>- mit Zeichen füllen</li><li>- Betragzahl auf Grenzen prüfen</li><li>- Festpunktzahl auf Grenzen prüfen</li><li>- Addieren, Vergleichen und Springen</li><li>- Byteadresse umsetzen</li><li>- einige Typ-D-Befehle (s. Arithmetiksimulationen)</li></ul> |

- 
- Arithmetik-  
simulationen Die Simulation der doppelt langen Festpunktarithmetik be-  
arbeitet auf der Zentraleinheit ZE01 Laden und Testen, Laden  
und Komplementieren, Vergleichen, Schieben und die vier  
Grundrechnungsarten, auf der Zentraleinheit ZE03 Multipli-  
zieren und Dividieren. Die Simulation der kurzen und doppelt  
langen Gleitpunktarithmetik bearbeitet Laden und Testen,  
Vergleichen und die vier Grundrechnungsarten.
- Konvertierun- Die nachstehend aufgeführten Funktionen sind mit Makros  
gen (s. 24 in /101/) und bis auf die Code-Umwandlung auch mit  
Anwender-Befehlen ansprechbar (s. /201/).
- . Dezimal/Be- Eine als Zeichenfolge vorliegende zulässige Dezimalzahl  
trag -> Dual oder Betragzahl wird in eine Dualzahl (16 bit) mit oder  
ohne Vorzeichen umgewandelt. Zulässige Dezimal- oder Be-  
tragszahlen entsprechen den Datentypen 'F' bzw. 'A' der  
Assembler-Sprache.
  - . Dual -> Eine 15-stellige Dualzahl mit Vorzeichen oder eine 16-  
Dezimal/ stellige Dualzahl wird in eine maximal sechs Zeichen lange  
Betrag Dezimalzahl mit Vorzeichen bzw. Betragzahl ohne Vorzeichen  
umgewandelt. Die Länge des Zeichenfeldes ist angebar. Bei  
fehlender Angabe werden für Dezimal- und Betragzahlen stets  
sechs Zeichen gewählt.
  - . Bitmuster Ein 16-stelliges Bitmuster wird in eine vier byte lange  
-> Hexa- Folge von Hexadezimalzeichen umgewandelt. Aus je vier auf-  
dezimal einanderfolgenden Bits entsteht ein Hexadezimalzeichen.
  - . Hexadezimal Eine maximal vier byte lange Zeichenfolge aus Hexadezimal-  
-> Bitmuster zeichen wird in ein 16-stelliges Bitmuster umgewandelt.  
Aus jedem Zeichen entsteht eine vier bit lange Dualzahl.  
Die Konvertierung endet beim ersten Zeichen, das kein He-  
xadezimalzeichen ist, oder nach vier Hexadezimalzeichen.  
Als Hexadezimalzeichen gelten die Ziffern 0 bis 9 und die  
Großbuchstaben A bis F.
  - . Code- Der Inhalt eines Zeichenfeldes wird von dem Siemens EBCDIC  
Umwandlung in den System-Code ASCII konvertiert und umgekehrt. Der  
Byteinhalt entspricht einem Platz in der Codetabelle  
(s. 40.4 in /101/). Die Code-Umwandlung erfolgt über den  
Lochkarten-Code, sie entspricht der eindeutigen Zuordnung  
eines Platzes des einen Codes zu einem bestimmten Platz des  
anderen Codes. Auf diese Weise werden auch Byteinhalte, die  
unbelegten Tabellenplätzen entsprechen, konvertiert.
- Hilfsroutinen Aus einem HSP-Feld (z.B. einem Eingabepuffer) wird ein Byte  
. Byte setzen gelesen. Es ist möglich, Zwischenräume zu überlesen oder  
sie als relevante Zeichen zu interpretieren.
- . Byte holen In eine HSP-Zelle (z.B. in einen Druckpuffer) wird ein Byte  
geschrieben. Das nicht angesprochene Byte der Zelle bleibt  
unverändert.  
Diese beiden Routinen werden durch Makros angesprochen  
(s. 24 in /101/). Ihnen fast gleich sind die Assemblerbe-  
fehle Byte speichern/laden (s. (201/)).
- . Tafelzeiger Zwei Systemsimulationen erlauben das Retten und Laden des  
kellern Tafelzeigerregisters TZR2 (nur Intern-Gebrauch, Befehle  
'H=E00A' und 'H=E00B').

---

## 8 Zeitbearbeitung

Das ORG stellt eine Reihe von Funktionen zur Zeitbearbeitung zur Verfügung, die unter der Voraussetzung eines Hardware-Zeitgebers im einzelnen folgende Leistungen bieten:

- ORG-Zeit** Das ORG führt intern eine "Software-Uhr". Der Benutzer kann sie als Datum und Uhrzeit lesen oder einstellen. Hierzu stehen ORG-Aufrufe und SBP-Kommandos zur Verfügung.
- Relativzeit** Neben der "Software-Uhr" führt das ORG zusätzlich einen Relativzeitzähler. Das Einstellen und Lesen der Relativzeit erfolgt über Aufrufe.
- Datumwechsel** Durch die Angabe des Datumwechsels bei der ORG-Generierung kann der Anwender den Zeitpunkt bestimmen, an dem der tägliche Wechsel des Datums erfolgen soll. Der Wechsel erfolgt stets zur vollen Stunde (s. 8.2).
- Zeitmodifizierte Aufrufe** Die Zeitbearbeitung ermöglicht dem Benutzer, bestimmte ORG-Aufrufe zeitverschoben oder zyklisch ausführen zu lassen. Das ORG führt dazu eine auf der Basis der Software-Uhr von ihm bearbeitete Zeitwarteschlange. Über ORG-Aufrufe oder SBP-Kommandos können diese Aufrufe wieder zurückgenommen werden.

### 8.1 Hardware-Einrichtungen

Die Zeitbearbeitung setzt einen Zeitgeber mit den folgenden Funktionen voraus:

- gepufferte Uhr,
- Kurzzeitwecker und
- periodischer Wecker

**gepufferte Uhr** Der Zeitgeber führt die Uhrzeit intern selbständig in Hardware-Zellen. Die Uhr kann vom ORG eingestellt und mit Millisekunden-Genauigkeit gelesen werden (s. 8.2).

**Kurzzeitwecker** Er wird durch Befehlsversorgung aufgezo- gen. Ist die Zeit abgelaufen, gibt er einen Interrupt an die ZE aus und geht dann in den Ruhezustand über. Das ORG benutzt ihn bei der Bearbeitung zeitmodifizierter Aufrufe.

**periodischer Wecker** Der periodische Wecker meldet der Zentraleinheit periodisch, daß die Weckzeit abgelaufen ist. Er wird wie der Kurzzeitwecker durch Befehlsversorgung aufgezo- gen. Das ORG benutzt ihn für die Zeitscheibenbearbeitung.

Der Zeitgeber hat eine gepufferte Stromversorgung, d.h. bei einem Ausfall der zentralen Stromversorgung wird weitergezählt. Er läßt sich durch Eingabe externer Minutenimpulse synchronisieren.

## 8.2 Software-Einrichtungen

Die Software-Einrichtungen der ORG-Zeitbearbeitung betreffen die Software-Uhr, den Relativzeitzähler, die Zeitaktualisierung sowie die Bearbeitung und Rücknahme zeitmodifizierter Aufrufe mittels Zeitwarteschlange.

**Software-Uhr** Auf der Basis des Zeitgebers führt das ORG eine Software-Uhr in 3 Wörtern des Software-VB mit folgendem Inhalt:

|        |                       |
|--------|-----------------------|
| Wort 0 | Minuten (höherwertig) |
| 1      | Minuten               |
| 2      | Millisekunden         |

Die Software-VB-Zellen werden nach Bedarf vom Zeitbaustein des ORG aktualisiert. Die Minuten zählen ab Bezugszeitpunkt 1.1.1972, 00.00 Uhr. Das ORG wandelt zwischen dieser internen Zeitdarstellung und der Kalender-Uhrzeit-Darstellung in Aufrufen, Kommandos und Meldungen um. Die Kalenderdaten werden bis zum Jahr 2071 verarbeitet.

**Datumseingabe** Die Software-Uhr wird über Aufruf oder SBP-Kommando auf volle Minuten eingestellt. Der Zeitgeber wird dabei auf 0 ms rückgesetzt; er zählt ab dieser Einstellung weiter. Ist Synchronisation eingestellt, werden mit dem nächsten externen Minutenimpuls Hard- und Software-Uhrzeit auf volle Minuten auf/abgerundet.

Die Uhrzeit kann vom Benutzer jederzeit neu eingestellt werden. Bei Batterieausfall am Zeitgeber ist dies zwingend. Zum Einstellen steht wahlweise ein Aufruf oder ein SBP-Kommando zur Verfügung. Der Aufruf ermöglicht es, das Datum auch zu einem fest vorgegebenen Zeitpunkt anzugeben. Hierbei ist vor allem an die Umstellung Sommer-/Winterzeit gedacht.

**Zeitaktualisierung** Die Software-Uhr "läuft" nicht kontinuierlich, das ORG aktualisiert sie von Zeit zu Zeit. Die Zeitaktualisierung betrifft die Fortschaltung der Software-Uhrzeit bzw. die Übernahme der Zeit aus den Hardware-VB-Zellen des Zeitgebers und die Neueinstellung von Datum und Uhrzeit durch SBP-Kommando oder Aufruf (s.o.). Es wird unterschieden zwischen automatischer Aktualisierung der Software-Uhr und Aktualisierung nach einem Spannungsausfall.

. automatisch Die Software-Uhrzeit wird in folgenden Fällen selbsttätig aktualisiert:

- Minutenalarm (Periodenweckeralarm)
- Übernahme zeitmodifizierter Aufrufe in die Zeit-WS
- Einstellen und Lesen des Datums (Aufruf oder SBP-Kommando)
- Wecker-Interrupt (Kurzzeit- und Periodenwecker)
- Einstellen und Lesen des Relativzeitzählers.



---

Beim Minutenalarm aktualisiert das ORG minütlich die Software-Uhrzeit im Abbild, bei entsprechender Generierung.

- . nach Netz- Bei Netzausfall enthalten die VB-Zellen der Software-Uhr  
ausfall den ungefähren Ausfallzeitpunkt. Sie werden von der Wiederanlaufoutine verwendet, um die Dauer des Ausfalls zu berechnen. Dazu wird der Inhalt der VB-Zellen in die Zellen der Ausfallzeit eingetragen.

Die Software-Uhr wird danach automatisch bei Spannungswiederkehr aktualisiert.

- Datumwechsel, Prozeßzeit-  
führung
- Normalerweise führt der Stundenüberlauf von 23.59 nach 0.00 zum Datumwechsel (Tag, Wochentag, ggf. Monat und Jahr). Prozeßanlagen mit Schichtbetrieb fordern oft eine Prozeßzeitführung, die den Datumwechsel zu anderen Zeiten, z.B. mit dem Schichtwechsel, ausführt.

Die Stundenangabe des Datumwechsels wird einer Generierungsangabe entnommen. Die Uhrzeit zwischen dem Datumwechsel und 0 Uhr bzw. zwischen 0 Uhr und Datumwechsel zählt dann schon zum neuen bzw. noch zum alten Kalendertag.

Die Zeitbearbeitung berücksichtigt die eingestellte Zeit des Datumwechsels bei allen absoluten Zeitangaben des Anwenders.

### 8.2.1 Zeitmodifizierte Aufrufe

Eine Reihe von Koordinierungsaufrufen erlaubt Modifikationen der Ausführungszeit. Solche Aufrufe werden, wenn sie beim ORG eintreffen, "aktiviert" (entgegengenommen), aber erst zu einem späteren Zeitpunkt "aktualisiert" (ausgeführt). Dies gilt auch für zugeordnete Kommandos.

Modifikationen Es sind folgende Zeitmodifikationen erlaubt:

- . AM Aufruf soll zur angegebenen Uhrzeit ausgeführt werden.
- . NACH Aufruf soll nach der angegebenen Verzögerungszeit ausgeführt werden.
- . ALLE Aufruf soll im Abstand der angegebenen Zykluszeit periodisch ausgeführt werden.
- . TAGL Aufruf soll jeden Tag zur angegebenen Uhrzeit ausgeführt werden.

Bearbeitung Zeitverschobene Aufrufe werden zu dem angegebenen Zeitpunkt nur einmal ausgeführt, danach nicht mehr. Der Aktualisierungszeitpunkt kann relativ (NACH) oder absolut (AM) angegeben werden. Zyklische Aufrufe werden mit der vorgegebenen Zykluszeit periodisch wiederholt (ALLE) bzw. täglich (TAGL).

**Aufrufe** Die folgende Tabelle gibt einen Überblick, welche Aufrufe mit welcher Zeitmodifikation gegeben werden können.

| Aufruf                              | AM | NACH | ALLE | TAGL |
|-------------------------------------|----|------|------|------|
| Anhalten eines Programms            | J  | J    |      |      |
| Datum einstellen                    | J  |      |      |      |
| Beenden eines fremden Programms     | J  | J    |      |      |
| Fortsetzen eines angehalt. Programm | J  | J    |      |      |
| Fortsetzen eines wartend. Programms | J  | J    | J    | J    |
| Uhrzeit einstellen für Monitor      |    | J    |      |      |
| Koordinierungszähler betätigen      | J  | J    | J    | J    |
| Starten eines Programmes            | J  | J    | J    | J    |
| Zeitm. Koordinierungsauftr. löschen |    | J    |      |      |
| Zeitm. Aufruf löschen               |    | J    |      |      |

**Zeitwarteschlange**

Das ORG baut eine Zeitwarteschlange aus den Elementen des Listenpools auf. In die Zeit-WS werden alle zeitmodifizierten Aufrufe in der Reihenfolge ihrer Aktualisierungszeitpunkte eingetragen. Aufrufe gleicher Aktualisierungszeit stehen in der Reihenfolge der Aktivierung in der Zeit-WS.

**Rücknahme zeitmodifizierter Aufrufe**

Für die Rücknahme zeitmodifizierter Aufrufe stehen dem Benutzer mehrere Aufrufe zur Verfügung, die folgendes veranlassen:

- Rücknahme aller zeitmodifizierten Aufrufe für einen namentlich anzugebenden Koordinierungszähler.
- Rücknahme eines bestimmten zeitmodifizierten Koordinierungszähleraufrufs.
- Rücknahme aller zeitmodifizierten Aufrufe für ein bestimmtes Programm; auch als SBP-Kommando.
- Rücknahme zeitmodifizierter Aufrufe eines bestimmten Typs für ein Programm.

### 8.2.2 Relativzeitzähler

Das ORG führt neben der Software-Uhr einen zweiten Zeitzähler, der dazu dient, schnell und einfach Zeitdifferenzen zu bilden und Zeitergebnisse aufzulösen. Der Relativzeitzähler zählt die Millisekunden als doppelt lange Betragzahl (32 bit). Das reicht aus für ca. 49 Tage, 17 Stunden. Bei Überlauf zählt er von 0 ab weiter. Der Anwender kann den Zähler mit einem Aufruf jederzeit und auf jeden Wert einstellen.

---

## 9 Bedien- und Dialogfunktionen

**Zusammenfassung** Für die Kommunikation mit dem ORG und den Anwenderprogrammen stellt das ORG dem Benutzer einen leistungsfähigen Bedienteil zur Verfügung. Der Bedienteil bearbeitet unterschiedliche Bedieneingaben und Aufrufe.

Alle Bedieneingaben werden zuerst dem Bedienteil zugeleitet, der die Weiterleitung an die jeweiligen Bedienungsempfänger übernimmt. Fehler, die dies verhindern, werden bereits vom Bedienteil erkannt und gemeldet.

Das Bearbeiten von Aufrufen erlaubt es, die Bedienungsabläufe zu steuern und zu koordinieren. Damit kann der Benutzer parallele Prozesse bedienen, Prozesse simultan bedienen oder einen einfachen Mehrbenutzerbetrieb betreiben.

Im AMBOSS-4-Betrieb verwaltet das Bediensystem sämtliche Eingabedaten von Dialoggeräten und aus Prozeduren (s. /500/). Die Bedienung wartender Programme geschieht mit der "programmierten" Bedienung (s. 9.5).

Nach Einzelaussagen gegliedert stehen folgende Systemleistungen zur Verfügung:

**Kommandosprache** Für die Bedieneingaben an das Standardbedienprogramm (SBP) wurde im Rahmen des Betriebssystem eine geeignete Kommandosprache geschaffen. Die Bedieneingaben (Kommandos) sind sprachähnlich aufgebaut; einzelne Parameter werden durch Blank oder eine Blankkette getrennt. Grundlage ist allgemein die englische Sprache.

**ORG-Bedienung** Sie richtet sich direkt an das ORG (s. 9.2). Der Benutzer kann damit:

- ORG-Meldungen quittieren,
- Geräte für den E/A-Verkehr zuteilen und
- Sitzungen für Fernbedienungen an- bzw. abmelden.

**SBP-Kommandos** SBP-Kommandos sind Bedieneingaben an das Standardbedienprogramm. Sie bieten die Möglichkeit, Systemleistungen auf einfache und komfortable Weise in Anspruch zu nehmen (s. 9.3).

**Anwenderbedienung** Die Bedienungen richten sich an bedienbare Dienst- und Hilfsprogramme, Übersetzer, Compiler und Anwenderprogramme (s. 9.4)

**Programmierte Bedienung** Außer der Dialogführung über ein Bediengerät kann auch ein Programm Bedieneingaben einem anderen Programm übergeben (s. 9.5). Das Zielprogramm kann sein:

- das ORG,
- ein Standardbedienprogramm oder
- ein Anwenderprogramm.

---

**Fernbedienung** Mit der Funktion Fernbedienung (Teleservice, s. /310/) kann ein Servicezentrum das ORG und die Anwenderprogramme eines Kunden aus der Ferne bedienen. Der Austausch von Bediendaten erfolgt hardwaremäßig über die Teleservice-Schnittstelle (s. /60/). Über sie kann der Betreuer mit dem Kundenrechner so arbeiten, als stünde im Servicezentrum ein Kunden-Terminal zur Verfügung.

Aus der Sicht des ORG-Bedienteils kann das Servicezentrum Systemleistungen nur dann ansprechen, wenn der Kunde den Zugang zur Fernbedienung freigegeben hat (s. 9.2). Der Bedienteil übernimmt dabei folgende Aufgaben (s. /103/):

- Remote-Schnittstellen des lokalen Teleservice-Anschaltmoduls (LTSA) freigeben/sperrern
- Programme für Teleservice freigeben/sperrern

**Zuteilen** Der Bedienteil unterstützt den Benutzer beim Koordinieren von E/A-Operationen. Beim Zuteilen wird ein physikalisches Gerät aus der Geräteliste des ORG einem Programm zugeteilt. Die Zuteilung kann wahlweise mit oder ohne Dialog erfolgen.

**Programm-Geräte-Zuordnung** Sie erlaubt die feste Zuordnung eines Bediengeräts zu einem bedienbaren Programm (Bedien-/Meldegerät). Dadurch ist ein einfacher Mehrbenutzerbetrieb ohne sonstigen Koordinierungsaufwand möglich (Terminalbetrieb s. 9.4).

**Bedienzeichenersatz** Der Bedienzeichenersatz (Kurzzeichenbedienung) erleichtert Bedieneingaben. Sie erlaubt es, eine Bedienung teilweise oder vollständig durch einen frei wählbaren Kurzzeichenstring zu ersetzen (s. 9.6).

## 9.1 Bedienungsablauf

Die folgenden Ausführungen erklären allgemeine Aspekte der Bedienung. Der Benutzer verkehrt mit dem ORG-Bedienteil über Bedieneingaben an Bediengeräten. Für Bedienungsübergaben zwischen Programmen wird die "programmierte Bedienung" (s. 9.5) benutzt.

**Bediengerät** Als Bediengeräte sind Datensichtstationen (DSS) zugelassen. Beim ORG-Generieren legt der Anwender fest, für welche dieser Geräte Bedienbarkeit zugelassen ist. Besondere Bediengeräte sind das Standardmeldegerät und die Bedien-/Meldegeräte.

**Standardmeldegerät** Eine DSS muß beim Generieren zum Standardmeldegerät (s. 13) erklärt werden. Zusätzlich kann eine andere DSS als Ersatzgerät dienen. Das Standardmeldegerät erlaubt die Bedienung aller Programme; während andere Bediengeräte nur bestimmte Programme bedienen dürfen (s.u. Grenzprogrammnummer). Durch einen Aufruf kann das Standardmeldegerät im laufenden Betrieb je nach Bedarf umdefiniert werden.

Bedien-/Meldegerät Mit einem Aufruf oder Kommando läßt sich eine DSS als Bedien-/Meldegerät einem oder mehreren Programmen fest zuordnen. Die Zuordnung ist änderbar. Der gesamte Dialog mit dem Programm (Bedieneingaben, Meldungen aller Art) geht ausschließlich über das zugeordnete Bedien-/Meldegerät. Dieses übernimmt insoweit die Aufgaben des Standardmelde- und Fehlermeldegerätes. Programm-Geräte-Zuordnung und Terminalbetrieb s. 9.4.

Grenzprogrammnummer, Bedienschutz Bestimmte Programme (z.B. das Betriebssystem) können vor Bedieneingaben von beliebigen Terminals aus geschützt werden. Dazu wird beim Generieren eine Grenzprogrammnummer genannt. Bedienungen für Programme bis zu dieser Nummer sind nur vom Standardmeldegerät möglich (auch nicht vom Bedien-/Meldegerät).

Bedienungseinleitung Der Bediener schreibt die komplette Bedieneingabe in eine Zeile des Bildschirms der DSS. Nach dem letzten Zeichen drückt er die DÜZ-Taste und ruft damit das ORG. Ist das Gerät für Bedienung zugelassen, so wird ein ORG-interner Standard-Eingabeaufruf abgegeben.

Bedieneingabe Je nach Bedienungsempfänger lassen sich die Bedieneingaben gliedern in:

|  | <u>Empfängeridentifikation</u> |
|--|--------------------------------|
| - ORG-Bedienungen, s. 9.2              | @                              |
| - SBP-Kommandos, s. 9.3                | /                              |
| - Anwenderprogramm-Bedienungen, s. 9.4 | :nr: oder :name:               |

Eine Bedieneingabe besteht aus der Empfängeridentifikation gefolgt von dem Bedientext. Sie endet mit dem Endezeichen ETX (wird bei Eingabe durch DÜZ-Taste erzeugt). Die Empfängeridentifikation ist eine Kennung, die den Bedienungsempfänger identifiziert. Sie ist dem Bedientext voranzustellen.

Bedienpuffer Für die Aufnahme des Bedientextes müssen Bedienpuffer zur Verfügung stehen. Es wird unterschieden zwischen Gerätepuffer im ORG und Anwenderpuffer.

Gerätepuffer Für jedes Bediengerät ist beim ORG-Generieren ein Gerätepuffer festzulegen. Systemvoreinstellung = 80 byte.

Anwenderpuffer Bedienbare Anwenderprogramme müssen ein Datenfeld zur Aufnahme des längsten zu erwartenden Bedientextes zur Verfügung stellen.

Sind die Puffer zu klein, wird der Bedientext abgeschnitten.

Fehlerfälle Der Bedienteil erkennt und meldet folgende Fehler:

- Die Bedieneingabe kann nicht weitergeleitet werden.
- Bei der Aufrufbearbeitung sind Anzeigen aufgetreten.
- Fehler bei der ORG-Bedienung.

Die Meldungen bestehen aus einer Identifikation, einer Kennziffer und dem Doppelkreuz (s. 4.3 in /103/).

Fehlermeldungen aufgrund von ORG-Bedienungen (s. 9.2) werden auf dem Standardmeldegerät ausgegeben. Besteht eine feste Zuordnung zwischen einem Bedien-/Meldegerät und einem Programm, so werden Meldungen, die das Programm betreffen, über das zugeordnete Bedien-/Meldegerät ausgegeben.

## 9.2 ORG-Bedienungen

ORG-Bedienungen richten sich direkt an das ORG. Der Benutzer kann damit:

- ORG-Meldungen quittieren,
- Geräte für den E/A-Verkehr zuteilen,
- Remote Schnittstellen für die Fernbedienung freigeben bzw. sperren.

**Bedieneingabe** Eine Bedieneingabe mit der Kennung @ identifiziert das ORG als Empfänger.

**Quittungen** Bestimmte Gerätefehler meldet das ORG mit einer Quittungsaufforderung. Diese Aufforderung beginnt mit dem Zeichen @ und endet mit dem Zeichen Q!. Die Aufforderung muß quittiert werden.

Bei AMBOSS 4 wickelt das Bediensystem (s. /500/) den Meldungs- und Quittungsverkehr mit dem ORG ab.

**Zuteilen** Beim Zuteilen wird ein Gerät aus der Geräteliste des ORG einem Programm zugeteilt. Bei der Zuteilung mit Dialog wird vom Bediener die Angabe eines logischen Gerätenamens verlangt.

**Fernbedienung** Eine Fernbedienung über die Remote-LTSA-Sichtgeräte-Schnittstelle (s. /60/) kann u.a. nur dann erfolgen, wenn sie über das ORG freigegeben wird (s. /103/).

Die Bedienungen "Schnittstelle freigeben" und "Schnittstelle sperren" werden nur über das Standardmeldegerät oder über die Remote-LTSA-Schnittstelle entgegengenommen. Nach der Schnittstellenfreigabe sind alle nicht explizit gesperrten Programme für Fernbedienung zugänglich.

## 9.3 Standardbedienprogramme

Das Betriebssystem stellt zwei Standardbedienprogramme (SBP, zusätzlich Mini-SBP, s. /103/) zur Verfügung. Sie gehören zum ORG-Masterstapel. Es sind jedoch selbständige Programme im Sinne des ORG, die auf Anwenderebene ablaufen. Sie ermöglichen dem Benutzer, eine Vielzahl von Systemleistungen in Anspruch zu nehmen. Die Anforderungen an das System formuliert der Benutzer mit Kommandos. Diese Kommandos werden geprüft, in ORG-Aufrufe umgesetzt oder vom SBP selbst ausgeführt. Kommandos gehen alternativ an SBP oder Mini-SBP.

SBP

Das SBP ist ein segmentiertes PRP. Bei fehlendem Bediensystem ist es beim Wiederanlauf durch das Wiederanlaufprogramm zu starten. Fehlt ein Wiederanlaufprogramm, so erfolgt der Start durch das ORG.

Bei AMBOSS 4 ist das SBP lediglich während des Systemstarts und nach Ausfall des Bediensystems erforderlich (s. /500/).

Das SBP kann vom Benutzer nicht geladen oder gelöscht werden. Die Bedienung erfolgt über das Standardmeldegerät oder dessen Ersatzgerät, nur wenn beim Generieren gewünscht, von allen bedienbaren Geräten (s. 9.1, Grenzprogrammnummer).

Mit den Kommandos werden

- Programm- und Zeitfunktionen ausgelöst sowie bestimmte Zuordnungen vereinbart;
- Lade- und Sicherungsfunktionen angestoßen, Geräte/Laufwerke unklar, offline oder online erklärt;
- Geräteparametrierdaten protokolliert/geändert;
- Bedieneingaben und Korrekturingaben vereinfacht, auf Batchbetrieb oder Monitorbetrieb umgeschaltet, bestimmte Protokollfunktionen veranlaßt, Meldungen übergeben;
- Systemdaten und Gerätedaten ausgegeben;
- Logbuchfunktionen;
- Wiederanlauffunktionen vereinbart oder angestoßen, Koordinierungszähler und Dateien eingerichtet oder gelöscht;
- Auskünfte über Vorbesetzungen, Kommandos und Geräte ausgegeben;
- Kommandos für Teleservice ausgeführt;
- auf HSP-Betrieb umgeschaltet und damit das Mini-SBP initiiert.

Mini-SBP

Das Mini-SBP ist ein HRP. Es wird vom SBP gestartet, wenn ein für den HSP-Betrieb generiertes ORG zum Ablaufen kommen soll. Es muß vom Benutzer geladen werden.

Entsprechend dem HSP-Betrieb besitzt das Mini-SBP gegenüber dem SBP einen kleineren Funktionsumfang. Mit Kommandos werden

- Programm- und Zeitfunktionen ausgelöst, Programm-/Geräte-Zuordnungen vereinbart;
- Ladefunktionen angestoßen, Geräte/Laufwerke offline oder online erklärt;
- bestimmte Protokollfunktionen angestoßen, Meldungen übergeben;
- Kommandos für Teleservice ausgeführt;
- auf PSD-Betrieb umgeschaltet und damit das SBP initiiert.

---

#### 9.4 Bedienbare Anwenderprogramme

Anwenderprogramme (HRP oder PRP), die Bedienungseingaben oder nur den Anruf eines Gerätes erwarten, geben einen Bedienaufruf an das ORG. Es sind zwei Modifikationen möglich: "Bedienungseingabe" und "Anruf erwarten". Der Bedienteil im ORG bearbeitet den Aufruf, ohne daß das Programm unterbrochen werden muß (Wartefunktion s.u.). Für jede Modifikation wird eine Warteschlange geführt.

"Bedienungseingabe" Mit dem Aufruf wird die Bedienbereitschaft für das aufrufgebende Programm freigegeben. Der Aufruf nennt ein Datenfeld, das den Bedienungstext aufnehmen kann.

"Anruf erwarten" Mit dem Aufruf kann (zwecks Koordinierung von Bediener und Programm) auf die Anruf-Betätigung eines bestimmten Gerätes gewartet werden. Es werden keine Daten übernommen.

Warten Zur Koordinierung zwischen Bedienaufruf und Programmablauf bestehen folgende Möglichkeiten:

- Bedienaufruf mit explizitem Warten,
- Bedienaufruf mit implizitem Warten,
- Bedienaufruf mit Mehrfachwarten und
- Bedienaufruf ohne Warten.

Im Falle der Wartefunktion wird das aufrufgebende Programm "wartend". Das ORG setzt es fort, wenn der Bedientext vollständig in das aufrufspezifische Datenfeld übertragen wurde oder ein Anruf (Anruftaste) an einem Bediengerät erfolgte.

Liegt ein Bedienaufruf ohne Warten vor, so wird das aufrufgebende Programm nicht unterbrochen. Eintreffende Bedienungen werden vom ORG simultan zum Programmablauf entgegengenommen.

In unter AMBOSS 4 (s. /500/) ablaufenden Programmen ist ein Bedienaufruf ohne Warten nur sinnvoll, wenn auf "Abbruchbedienungen" reagiert werden soll.

Räumbarkeit Auf Bedienung wartende PRP sind jederzeit räumbar. Eintransfer und Fortsetzung des Programms richten sich nach den Laufbereichsbedingungen und Prioritäten.

Mehrere Bedienaufrufe Ein fortgesetztes Programm (das gerade die Bedieneingabe bearbeitet) steht nicht in der Bedienungswarteschlange. Eine weitere eintreffende Bedienung wird deshalb abgewiesen. Dies kann durch mehrere Bedienaufrufe (z.B. zwei) und zugeordnete Wartefunktion verhindert werden.

Es ist vom ORG sichergestellt, daß nach erfolgter Bedienung und Programmfortsetzung die Zuordnung des Bedienungsvorgangs zum Datenfeld des Bedienaufrufs in chronologisch richtiger Reihenfolge gegeben ist.



---

Mehrbenutzer- Mehrbenutzerbetrieb bedingt u.a. eine feste Zuordnung eines betrieb Bediengerätes zu einem bedienbaren Programm. Hierfür gestattet das ORG, Terminalbetrieb zu fahren. Der gesamte Dialog (Bedieneingaben, Meldungen) mit einem oder mehreren Programmen geht über ein zugeordnetes Terminal; andere Terminals können den Programmbetrieb nicht stören. Das ORG realisiert dies in zwei Funktionsgruppen:

- Terminalbetrieb
- Zuordnen von Programm zu Gerät

Terminal- Der Terminalbetrieb wird am zuzuordnenden Terminal durch betrieb ein Kommando eingestellt und wieder aufgehoben (/LOG., s. 3.1.13 in /103/). Im Terminalbetrieb ist jedem vom Terminal gestarteten Programm bis zum Programmende das Terminal als Bedien-/Meldegerät zugeordnet.

Das ORG kann max. 20 zugeordnete Terminals parallel verwalten. Die Anzahl zugeordneter Programme ist nicht begrenzt, auch nicht die Anzahl der e i n e m Gerät zugeordneten Programme.

Nach einem Wiederanlauf besteht keinerlei Zuordnung; der Terminalbetrieb ist dann aufgehoben.

Programm- Außer dem Terminalbetrieb ist es möglich, Zuordnungen ein- Geräte- zeln zwischen einem Programm und einem Gerät (Bedien-/Melde- Zuordnung gerät zu treffen und wieder aufzuheben. Die Funktion "Programm-Geräte-Zuordnung" wird durch einen Aufruf (s. 21.6 in /101/) oder ein Kommando (s. 3.1 in /103/) angestoßen. Das Programmende, ein entsprechendes Kommando, ein entsprechender Aufruf oder ein Wiederanlauf heben eine bestehende Zuordnung auf.

Besteht die Zuordnung mit einem Gerät, das virtuell, global oder nichtbedienbar erklärt ist, so kann das Programm auch von anderen Geräten bedient werden.

## 9.5 Programmierte Bedienung

Ein Programm kann Standardtext einem anderen Programm, das diesen als Bedienung erwartet, übergeben. Der Aufruf "Programmierte Bedienung" im Sendeprogramm übernimmt diese Aufgabe. Das Zielprogramm muß diese Daten in Form eines Bedienauffrufes erwarten.

Bei AMBOSS 4 übergibt das Bediensystem (s. /500/) die Bedienungen an die Anwenderprogramme oder an das ORG.

Bedienungs- Im Aufruf wird das Zielprogramm (direkt oder indirekt), übergabe ein Textfeld und wahlweise ein Beifeld angegeben. Das Textfeld enthält die Bediendaten.

Bei der direkten Angabe des Zielprogramms wird ab Textfeldanfang übertragen. Nur zulässig bei Anwenderprogrammbedienung.

Bei indirekter Angabe des Zielprogramms wird ab dem Byte nach der Empfängeridentifikation übertragen. Das Textfeld ist dann so aufgebaut wie der Text einer Gerätebedienung (Empfängeridentifikation, Bedienungstext und -abschluß). Aus der Identifikation ermittelt das ORG das Zielprogramm und überträgt den Text ohne die Identifikation.

Das aufrufende Quellprogramm läuft erst nach abgeschlossener Bedienungübertragung weiter. Quell- und Zielprogramm können PRP sein und auch zum gleichen Laufbereich gehören.

**Fehlerfälle** Fehlt im zu übertragenden Text das ETX, wird bis zum Feldende (des Quellprogramms) übertragen, ohne Anzeigen zu übergeben, sofern das Feld im Zielprogramm lang genug ist.

### 9.6 Bedienzeichenersatz

Zur Vereinfachung von Bedienungseingaben besteht die Möglichkeit, häufig wiederkehrende Bedienungen durch einen Kurzzeichenstring zu ersetzen. Er wird einmal der vollständigen Bedienungszeichenfolge zugeordnet und kann ab diesem Zeitpunkt statt der Bedienungszeichenfolge benutzt werden. Die Kurzzeichenbedienung kann von einem Gerät oder programmiert erfolgen. Sie setzt den Gleitpunktprozessor oder den Generierwunsch "SIMFPL" voraus.

**Zuordnung** Der frei wählbare Kurzzeichenstring besteht aus 1 bis 6 abdruckbaren Zeichen. Die Zuordnung zwischen Kurzzeichenstring und Bedienungszeichenfolge wird über ein SBP-Kommando getroffen (s. 3.3 in /103/).

Die ersetzbare Bedienungszeichenfolge kann eine vollständige Bedienung (Identifikation und Text) oder ein ununterbrochener Teil davon sein. Zwischenräume bleiben erhalten. Die Folge ist auf 104 Zeichen beschränkt. Sie darf Zeichenersatzaufrufe enthalten (s. unten) oder aus solchen bestehen; dadurch ist eine Schachtelung möglich.

**Anwendung** Bedienzeichenersatz kann verwendet werden bei

- Bedienungseingaben
- Standardbedienungen als Kommandos (nicht im Batchbetrieb)
- ORG-Bedienungen
- programmierten Bedienungen

Die Verwendung des Bedienzeichenersatzes erfolgt im Zeichenersatzaufruf, dargestellt als eingeklammerte Kurzzeichenfolge.

Zeichenersatzaufruf: kla kurzzeichenstring kle

Die beiden Klammerzeichen kla und kle wählt der Anwender beim Generieren; sie sind vorbesetzt mit [ und ] . Zwischenräume sind zwischen den beiden Klammerzeichen nicht erlaubt.

10 Generierfunktionen

System-  
 erstellung

Das Systemsoftwareangebot zu den Rechnersystemen SICOMP M/ 6.680 unterstützt verschiedenartige Erstellungsmöglichkeiten für ein Anwendersystem. Die - abhängig von Anwendungsbereich und Erstellungsaufwand beim Anwender - in Bild 10/1 angegebenen Alternativen gestatten optimales Vorgehen des Anwenders.

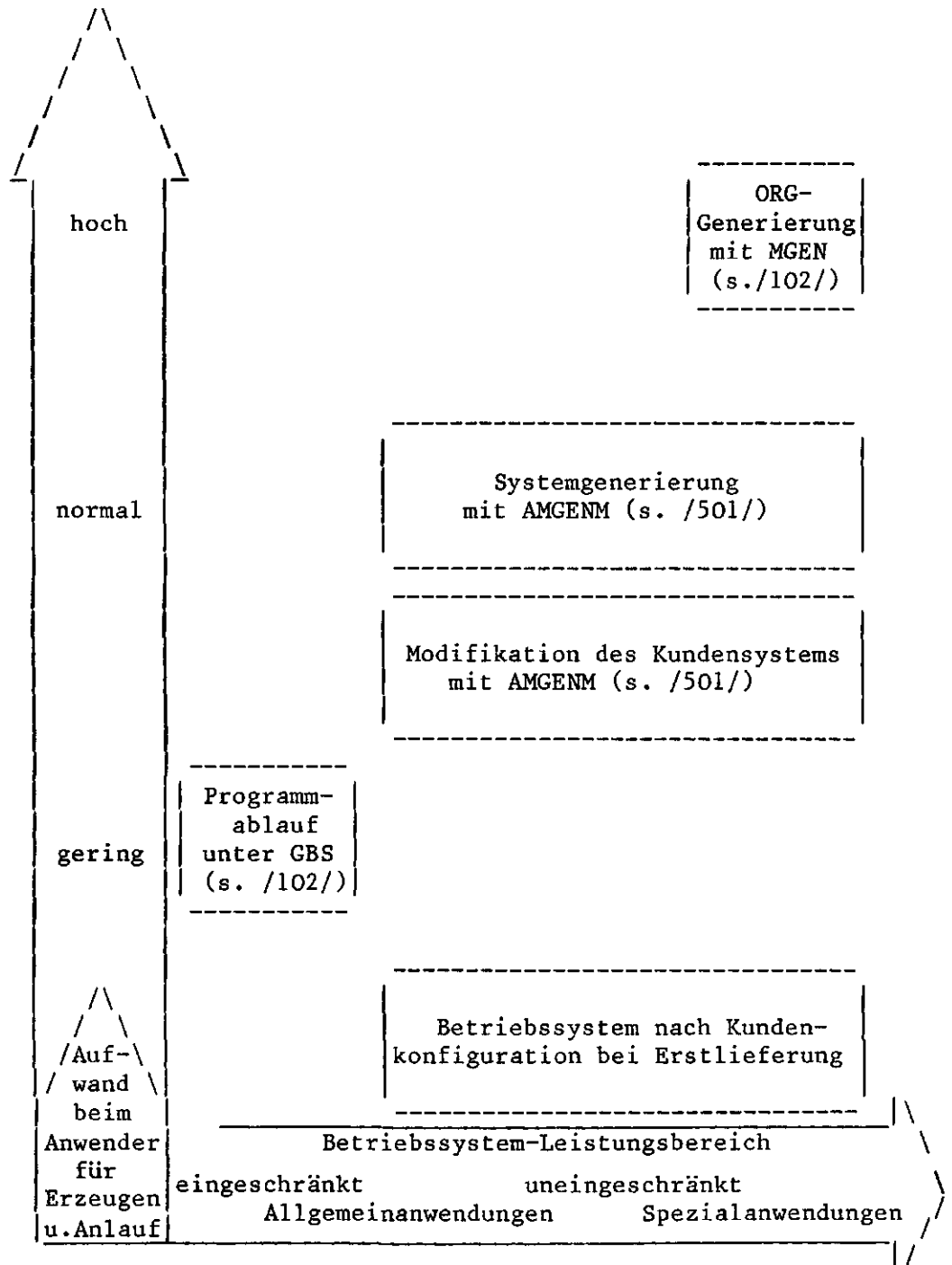


Bild 10/1: Erstellungsmöglichkeiten für ein Anwendersystem

- 
- ORG-Generieren Für den Anwender ist das in diesem Abschnitt beschriebene Generieren auf den Zentraleinheiten ZEO1, ZEO2, ZEO3 und ZEO4 (s. /25/, /26/) mit dem ORG-Generator MGEN nur von Bedeutung, wenn der Einsatz des AMBOSS-Generators AMGENM nicht möglich ist (AMGENM, s. /501/).
- Lieferform, Lieferumfang Es werden die gelieferten Magnetbandkassetten zugrunde gelegt. Sie enthalten u.a. die zum Generieren benötigten Programme Grundbetriebssystem, ORG-Generator, den gemeinsamen ORG-Masterstapel (einschließlich DVS-M und SPOOL-M) sowie die Dienstprogramme INITM und COPY-M im Abbild im Urladeformat.
- Die Magnetbandkassetten enthalten auch Standardbedienprogramme, Lader und auf Bestellung ein anwendungsspezifisches Betriebssystem.
- Grundbetriebs- Der ORG-Generator und die Programme INITM und COPY-M setzen system (GBS) zum Ablauf ein ORG voraus. Dieses ORG steht in Form des Grundbetriebssystems (GBS) als kompaktes HSP-residentes ORG fester Funktionsausstattung zur Verfügung.
- ORG-Varianten Man kann auf einem Datenträger mehrere ORG für die eigene Anlage oder für fremde Anlagen erzeugen. Wünscht der Anwender mehrere ORG-Varianten aus betrieblichen Gründen alternativ zu betreiben, so muß er sie mit unterschiedlichem Systemkennzeichen (das in den ORG-Namen eingeht) generieren, auf PSD ablegen und per Wiederanlauf unter Nennung des ORG-Namens aktivieren.
- Generier-varianten Je nach Generiervoraussetzungen und Randbedingungen für den Systemdatenträger sind unterschiedliche z.T. modifizierbare Generiervarianten möglich, nämlich
- Generieren mit bzw. ohne Systemkennzeichen-Prüfung,
  - Probegenerieren mit bzw. ohne Protokoll,
  - Generieren eines Korrektursystems im Rahmen der sogenannten "Änderungsgenerierung",
  - ORG-Löschen,
  - HSP-ORG vorbereiten.
- Systemkennzeichen-Prüfung Da jeder Datenträger Systemspeicher für mehrere ORG sein kann, hat der Anwender beim Generieren die Möglichkeit, bereits bestehende Systemdateien/Systeme mit demselben Systemkennzeichen vor ungewollter Zerstörung zu schützen.
- Probegenerieren Mit dieser Generiervariante erhält der Anwender wichtige Informationen über das zu generierende ORG, ohne es zu erzeugen. So werden ihm beispielsweise aufgetretene Generierfehler gemeldet und freier Speicherplatz berechnet. Diese Generiervariante benötigt 50 Prozent der normalen Generierzeit.
- Darüber hinaus führt die Modifikation "Probegenerieren ohne Protokoll" keine Prüfung des Masterstapels auf korrekte Externdefinitionen durch. Sie benötigt 20 Prozent der normalen Generierzeit.

- 
- ORG-Löschen** Mit dem ORG-Generator ist es möglich, auf einem Datenträger bis zu 38 unterschiedliche ORG zu erzeugen. Mit der Generiervariante "ORG-Löschen" kann der Benutzer einzelne ORG wieder vom Systemdatenträger entfernen.
- HSP-ORG-Vorbereiten** Der ORG-Generator kann bei entsprechenden Anwenderangaben ein ORG vorbereiten, das nach Installation und Inbetriebnahme keine systembedingten Peripherspeicherzugriffe mehr benötigt.
- Änderungsgenerierung** Als Änderungsgenerierung bezeichnet man den unter einem bestehenden Anwendersystem ablaufenden Austausch des zugrundeliegenden ORG.  
Es kann also unter dem laufenden Anwendersystem das ORG nach individuellen Wünschen des Systemerstellers neu konfiguriert werden, sofern gewisse Einschränkungen bezüglich der Größe von Datenadreßräumen beachtet werden. Anschließend wird das gesamte Anwendersystem auf das "neue" ORG aufgesetzt. Nach erfolgter Änderungsgenerierung sind somit sämtliche Komponenten eines AMBOSS-4-Systems bzw. BS-M-Systems im Sinn des "neuen" ORG wieder ansprechbar (z.B. Systemprogramme, Dienstprogramme, Anwenderprogramme, Standardprozeduren, Benutzer im Sinn des Betriebssystems BS4 von AMBOSS 4).
- Im folgenden sind einige wichtige Anwendungsfälle der Änderungsgenerierung angegeben:
- Änderung der Hardware-Konfiguration eines AMBOSS-4-Systems bzw. BS-M-Systems ohne Neugenerierung des Anwendersystems.
  - Einbringen eines neuen ORG-Ausgabestands/einer neuen ORG-Version in ein bestehendes Anwendersystem ohne Neugenerierung des Anwendersystems.
  - Komfortable Fremdgenerierung bzw. Testgenerierung mittels AMGENM.  
In diesen beiden Anwendungsfällen soll ein Anwendersystem für eine (Ziel-)Anlage der Modellreihe SICOMP M bzw. 6.680, das auf einer Erstellungsanlage SICOMP M bzw. 6.680 anderen Typs bzw. mit einer anderen Hardware-Konfiguration erstellt wurde, an die Hardware-Konfiguration der Zielanlage angepaßt werden. Bei der Testgenerierung wird zusätzlich ein Systemtest an der Erstellungsanlage vorgenommen (Details s. /501/).
  - Neudimensionierung von diversen ORG-Listen bzw. der Laufbereiche unter dem bestehenden Anwendersystem unter Berücksichtigung der vorgegebenen Restriktionen bezüglich der Größen von ORG-Adreßräumen.

✓

✓

✓

✓

---

Die Änderungsgenerierung besteht aus zwei Arbeitsschritten, die vom Anwender nacheinander angestoßen werden müssen:

- o der Generierung eines Korrektursystems (auch "Änderungssystem" genannt) mit Hilfe des ORG-Generators MGEN und
- o der Übertragung des gesamten Anwendersystems in das Korrektursystem (also zum "neuen" ORG), dem sogenannten "Änderungslauf" mit Hilfe des Dienstprogramms MODGEN.

Details zur Änderungsgenerierung s. /102/!

**Laden** Der ORG-Generator läßt sich durch den Lader des aktuellen ORG laden. Der Platzbedarf des ORG-Generators beträgt ca. 42 Seiten. Der ORG-Generator läßt sich als HRP oder PRP (auch mit V-Teil-Verlängerung) laden und ist wiederstartbar.

**Datenträger** Die Generierung setzt PSD voraus und legt auf ihnen Systemdaten ab. So betriebene Geräte müssen dem benutzten ORG (z.B. GBS, s.o.) bekannt, die zugehörigen Datenträger eingerichtet sein.

**Systemdateien** ORG-Generator und ORG verwalten alle Systemdaten auf PSD in Form von Bibliotheken und/oder Dateien. Der Generator richtet auf angebbaren ggf. verschiedenen PSD eine Reihe von Systemdateien ein. Sie sind weitgehend durch Anwender-Generierparameter bedingt, nämlich:

- Systemabbild
- Systemkontrolldaten
- Bedienzeichenersatz
- ORG-Rettbereich
- Abbild für HSP-ORG
- Informationen vom Probegenerieren
- ORG-Segmente
- peripherspeicherresidente Programme
- Pufferzonen für Datenpufferung
- Anwenderrettbereiche

Durch vom Anwender generierte ORG-Subsystem-Schnittstellen können weitere Systemdateien entstehen.

**Systemdatei-  
verwaltung** Werden PSD häufig wechselweise für verschiedene Generierungen benutzt, muß der Anwender ggf. die Platzverwaltung unterstützen (s. ORG-Löschen).

**Generier-  
parameter** Der ORG-Generator leistet die Generierung gemäß den vom Anwender anzugebenden Parametersätzen. Diese beschreiben Anlagenausstattung, Dateiorte, Platzbedarf, Buchführungsanforderungen, Anfangsparameter usw. wie folgt:

- Arbeitsformspeicher für segmentierte Programme
- Abbild- und Quellspeicher, Systemkennzeichen, HSP-ORG, Adreßraumerweiterung, Anzahl der ORG-Adreßräume
- Rettspeicher, Rettbereiche
- ORG-Segmentspeicher
- Anwender-Anlauf-/Wiederanlaufprogramm (PRP)

- Maximale Objektanzahl
- Begrenzungszeichen und Dateilänge für Bedienzeichenersatz
- Pool-Elementanzahl
- Generierprotokoll, Protokollgerät
- Definition der Paßwörter für virtuelle Konsole und Teleservice
- Prioritäten für Standardbedienprogramm, Grundsprache-Lader und Arbeitsform-Lader
- Grenzprioritäten für die Umrechnung der Software-Prioritäten
- Standardmeldegerät, Grenzprogrammnummer
- Bausteinwünsche
- Definition alternativer Laufbereiche und der maximalen Laufbereichsanzahl
- Länge des Commonbereichs
- Definition der Länge der Externlistendatei für Pakete, Common-Bereiche und Common Code
- HSP-Ausbau, Batteriepufferung
- Koordinierungszähleranzahl
- Fixe selbständige Laufbereiche und Paket-Laufbereiche
- Programmpakete
- Speicher für peripherspeicherresidente Programme
- Logische/physikalische Geräte
- Definition von Geräte-Parametrierdaten und/oder OSI-Schichtadressen
- Datenpufferungsgeräte (L-Modus), Puffer
- Datenpufferungsgeräte (S-Modus), Puffer
- Pufferzonendatei, Datenpufferung, S-Modus
- Arbeitspuffer für Dateiverwaltung
- Definition eines Elementes für Batchbetrieb
- Definition der logischen LTSA-Gerätenamen
- Definition der Logbuchführung
- PROMEA-Zeitgeberparameter
- Definition (Einrichten) einer Feldeinheit
- Definition eines Feldbereichs
- Elementanzahl der DVS-Gerätetafel
- DUMMY-Schlüsselwert für DVS-M
- SPOOL-Speicher, -Funktion, Liste virtueller Geräte
- SINEC-M: Funktionsverbund, Liste globaler Geräte
- Definition eines PAGE-Kachelbereichs
- Maximale Anzahl der PAGE-Kachelbereiche
- LINK-Depot-Dimensionierung
- Systemspeicherwechsel
- Cachebetrieb, -bereiche
- Hilfsdateispeicher
- KS100-Parametrierung, -Listenbereichslänge

Aktivierung/  
Generierung

Mit Hilfe der virtuellen Konsole ist zunächst das Grundbetriebssystem durch Umladen von der Magnetbandkassette zu aktivieren. Nach dem Erstellen eines Bibliothekselements oder einer Datei mit den Generierparametersätzen ist der ORG-Generator zu laden und das gewünschte System zu generieren.



- 
- Fehler            Als Fehlerreaktionen sind STOP-Zustände und Meldungen am Bediengerät möglich.
- Bricht ein Generiervorgang infolge zu großer ORG-Objekte ab, so sind zur weiteren Vorgangsweise die Hinweise zur Adreßraumerweiterung zu beachten (s. 1.1).
- Einbringen von ORG-Korrekturen    Im Rahmen der Systemwartung vom Hersteller (auf einem geeigneten Transportdatenträger) gelieferte ORG-Korrektur Elemente (sogenannte "ORG-Korrekturbalkone") können mit Hilfe der Wartungshilfe EVITAM /120/ komfortabel in das laufende ORG eingebracht werden. Dazu ist keine Neugenerierung des Anwendersystems erforderlich!

✓

✓

✓

✓

## 11 Systemaktivierungsfunktionen

Das ORG stellt für die erste und spätere Inbetriebnahme (z.B. nach Systemwechsel, Störungen u.ä.) des Rechnersystems Grundleistungen zur Verfügung, die der Benutzer nach Bedarf ergänzen oder abwandeln kann.

System inbetriebnehmen s. /102/, System einschalten s. /10/.

**Wiederanlauf** Wiederanlauf bedeutet vorwiegend die Reaktivierung eines auf dem Peripheriespeicher mit Sicherheit intakten Systems. Der Sonderfall des ORG-Anlaufs wird nicht für sich betrachtet, weil aus ORG-Sicht kein Unterschied zum Wiederanlauf besteht.

**Neustart** Der Neustart setzt ein im HSP präsentenes intaktes Anwendersystem voraus und besteht aus dessen Rücksetzen in einen definierten Anfangszustand.

**Wiederaufsetzen** Wiederaufsetzen besteht im Fortsetzen eines Programms nach Spannungswiederkehr an der durch Spannungsausfall am ZE-Rahmen unterbrochenen Stelle. Der Systemzustand bleibt über die Unterbrechung hinweg erhalten.

**Übersicht** Bild 11/1 zeigt in erster Annäherung, in welchem Systemzustand die o.g. Funktionen durch welche Ereignisse ausgelöst werden können. Bei der Systemaktivierung durch Spannungswiederkehr ist mit Symbolen angedeutet, welche Funktion ausgelöst wird, wenn der Anwender den HSP batteriegepuffert betreibt (Inhalt erhalten "+") oder wenn er keine Batteriepufferung installiert hat (Inhalt verloren "-"). "ZE" deutet an, daß nur Spannungsausfall am ZE-Rahmen bei batteriegepuffertem Hauptspeicher zum Wiederaufsetzen führt (s.o.).

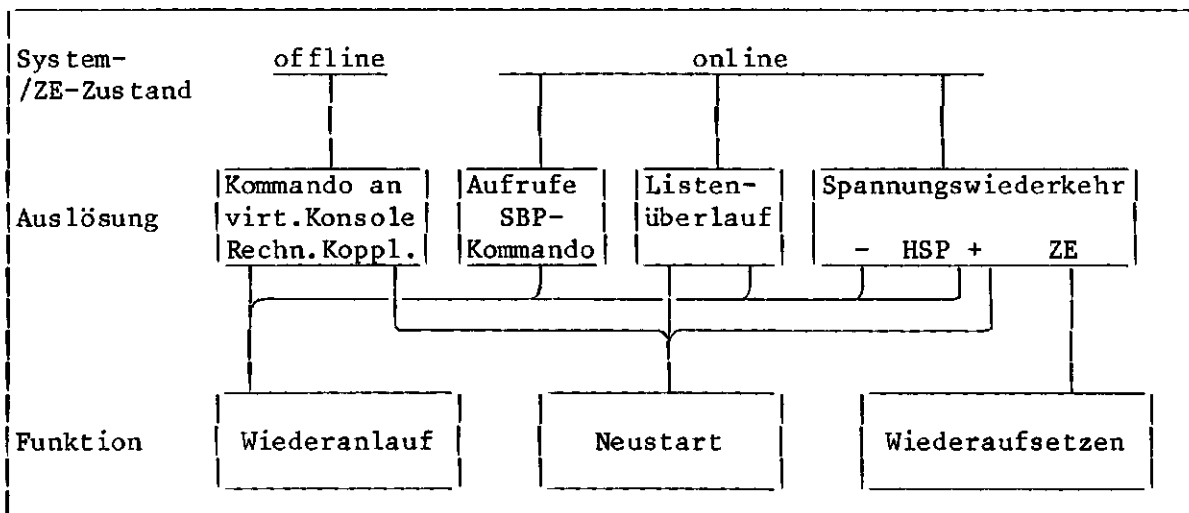


Bild 11/1: Übersicht Systemaktivierung

|                       |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |  |  |   |
|-----------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|--|---|
| Auslösung/<br>Ursache | Neustart von virtueller Konsole   | X |   |   |   |   |   |   |   |   |   |   |   |   |   |  |  |  |   |
|                       | Wieder-<br>anlauf über } virtuelle Konsole<br>} Aufruf                  |   | X |   |   |   |   |   |   |   |   |   |   |   |   |  |  |  |   |
|                       | Spannungs-<br>wiederkehr am } Erweiterungs-Rahmen<br>} ZE-/Grund-Rahmen |   |   |   | X | X |   | X | X | X | X |   |   |   |   |  |  |  |   |
|                       | Listenüberlauf<br>Urladeanstoß über Rechnerkopplung                     |   |   |   |   |   |   |   |   |   |   |   | X | X |   |  |  |  | X |
| Bedingungen           | HSP-Batteriepufferung   | / | / | / | / | / | N | J | J | J | / | / | / |   |   |  |  |  |   |
|                       | Generierwunsch "Neustart"   | J | / | / | N | J | / | N | J | N | J | N | J | N | / |  |  |  |   |
|                       | Generierwunsch "Wiederaufsetzen"  | N | / | / | / | N | / | J | N | N | N | N | / | / |   |  |  |  |   |
| Aktion                | Neustart  | X |   |   |   | X |   |   | X |   | X |   | X |   |   |  |  |  |   |
|                       | Wiederaufsetzen   |   |   |   |   |   |   | X |   |   |   |   |   |   |   |  |  |  |   |
|                       | Wiederanlauf  |   | X | X | X |   | X |   |   |   | X |   | X | X | X |  |  |  |   |

Tabelle 11/2: Voraussetzungen zur Systemaktivierung

In Tabelle 11/2 sind in Form einer Entscheidungstabelle die auslösenden Ereignisse in Verbindung mit maßgebenden Parametern (Bedingungen) dem zu erwartenden Systemverhalten (Aktionen) gegenübergestellt.

### 11.1 Wiederanlauf

**Anlaß** Störungen im Betrieb der Rechneranlage, gestörte Betriebsabläufe im Anwendersystem oder die Absicht, in einer neuen Arbeitsphase mit einem anderen Software-System zu arbeiten, bedingen in der Regel einen Wiederanlauf mit definierten Ausgangszuständen. Er wird - je nach Ursache - vom Anwender oder vom ORG eingeleitet.

ORG-Wiederanlauffunktionen lassen sich durch Anwendermaßnahmen ergänzen (Wiederanlaufprogramm). Ein Generieren der Wiederanlauf funktion oder Ladevorgänge sind nicht erforderlich. Das System ist umgehend wieder funktionsfähig.

**Teilschritte** Die Wiederanlaufphase setzt sich aus folgenden Teilschritten zusammen

- Wiederanlaufauslösung (s.u.)
- Festhalten der Wiederanlaufursache
- Retten des HSP-Inhaltes
- Einlesen des Abbildes
- Basisparametrierung, Parametrierung der Geräte
- Datenträgerüberprüfung
- Maßnahmen für Systemspeicherwechsel
- Aufbereiten und Aktivieren des ORG
- Start des Standardbedienprogramms, Startlistenbearbeitung bzw. Start des Wiederanlaufprogramms

Auslösung Ein Wiederanlauf läßt sich durch eines der folgenden fünf Ereignisse auslösen

- Kommando "Wiederanlauf" (über virtuelle Konsole)
  - Spannungswiederkehr
  - Listenüberlauf im ORG
  - Aufruf "Wiederanlauf"
  - Urladeanstoß über Rechnerkopplung
- } (kein "Neustart" generiert)

Der erste Fall sowie Spannungswiederkehr ohne Batteriepufferung und Urladeanstoß über Rechnerkopplung setzen kein intaktes ORG im HSP voraus. In den übrigen Fällen müssen dagegen die für den Wiederanlauf nötigen ORG-Teile im HSP noch funktionsfähig sein.

Systemmitteilung Bestimmten Anwenderprogrammen kann vom ORG auf Wunsch eine Anzahl von Wiederanlaufinformationen als Kennbits (ein Wort) übergeben werden. Das ist für ein Anwender-Wiederanlaufprogramm gedacht, das z.B. abhängig von der Wiederanlaufursache reagieren soll. Bei Benutzung der Wiederanlaufleistungen des Standardbedienprogramms wird ein Teil der Wiederanlaufinformation als Meldung ausgegeben.

Datenträger-Prüfungen Als Sicherheitsmaßnahme gegen Bedienungsfehler prüft das ORG bei jedem Wiederanlauf die bei der Generierung angegebenen Datenträger (PSD). Änderungen an ihnen (Wechselplattenaustausch, geänderte Datenträgerbuchführungen) werden erkannt. Soweit sie Prozeßdaten betreffen, können sie zu unübersehbaren Folgeschäden führen, wenn nach einem Wiederanlauf mit falschen Daten weitergearbeitet wird. Der Anwender wird über Wiederanlaufinformationen benachrichtigt. Datenträgerwechsel im laufenden Betrieb ist über Kommandos zu koordinieren.

Wiederanlaufmodi Ein Wiederanlauf ist durch eine Modusangabe in zwei Punkten modifizierbar. Sie betreffen Ausführen/Unterlassen von Startlistenaktivierung und Retten des HSP-Inhaltes.

Systemspeicherwechsel Mit der Funktion "Systemspeicherwechsel" erhält der Anwender die Möglichkeit, beim Ausfall des Laufwerks mit dem Abbildspeicher (im folgenden "Originallaufwerk" genannt) bestimmte logische Geräte dieses Laufwerks auf einem parallelen "Ersatzlaufwerk" mit unveränderten logischen Gerätenummern zu betreiben.

Der Systemspeicherwechsel ist in zwei Ausbaustufen verfügbar:

- der Grundfunktion und
- der erweiterten Funktion.

Bei Verwendung der Grundfunktion tauscht das ORG die logischen Gerätenummern des Abbildspeichers und des entsprechenden "parallelen" Plattenspeichers im "Ersatzlaufwerk", d.h. der Abbildspeicher bleibt auch im "Ersatzlaufwerk" mit seiner bisherigen logischen Gerätenummer ansprechbar.

✓

✓

✓

✓

---

Ein derart generiertes System ist in seiner Umladefähigkeit nicht mehr an die bei der Generierung festgelegte logische Gerätenummer des Abbildspeichers gebunden; das System läßt sich dann vom entsprechenden "parallelen" Plattenspeicher im "Ersatzlaufwerk" urladen. Diese Strategie ist z.B. dann von Vorteil, wenn das Laufwerk mit dem Abbildspeicher defekt wird.

Bei Verwendung der erweiterten Funktion können zusätzlich bis zu 31 weitere logische Geräte vom "Originallaufwerk" mit dem Abbildspeicher auf das "Ersatzlaufwerk" getauscht werden; sie sind dort ebenfalls mit ihren bisherigen logischen Gerätenummern ansprechbar. Diese Vorgangsweise erhöht ganz allgemein die Übersichtlichkeit bezüglich der logischen Gerätenummern bei Verwendung des Systemspeicherwechsels und ist z.B. dann vorteilhaft, wenn die Spooldatei und/oder die Arbeitsformen der Programme nicht auf dem Abbildspeicher angelegt wurden.

Voraussetzungen für den Systemspeicherwechsel:

- die Funktion "Systemspeicherwechsel" muß im gewünschten Leistungsumfang generiert sein
- die Produktnummern von "Originallaufwerk" und "Ersatzlaufwerk" müssen übereinstimmen
- die Kapazität der jeweiligen logischen Geräte, ihre Adreßzusätze sowie ihre logische Struktur müssen übereinstimmen
- auf den Plattenspeichern des "Ersatzlaufwerks" dürfen sich keine Systemdateien befinden
- alle Programme, die mit einem der getauschten logischen Geräte verkehren, sind in definierte Anfangszustände zu bringen
- der erfolgte Systemspeicherwechsel muß mit einem Wiederanlauf abschließen
- die Verwendung der erweiterten Funktion des Systemspeicherwechsels ist in Verbindung mit den AMGENM-Generierarten "Fremdgenerieren" bzw. "Testgenerieren" (s. /501/) sowie mit der Änderungsgenerierung (s. /102/) verboten.

Für genauere Informationen zum Systemspeicherwechsel wird auf die Handbücher /102/ und /501/ verwiesen.

## 11.2 Neustart

**Zielsetzung** Der Neustart stellt eine Alternative zum Wiederanlauf zwecks schneller Systemreaktion nach Störungen durch Spannungseinbrüche oder Listenpoolüberlauf dar. Ferner ist Rücksetzen des HSP ohne PSD-Zugriff möglich.

**HSP-ORG** Das Hauptspeicher-ORG wird teilweise aus Sicherheitsgründen eingesetzt. Der Neustart ermöglicht in Verbindung mit geeigneten Anwendermaßnahmen bei einem voll funktionsfähigen ORG, dieses nach einer Systemstörung auch ohne PSD-Zugriff kurzzeitig in einen definierten Anfangszustand zurückzusetzen und ggf. einen sicheren Betrieb aufrechtzuerhalten. Retten des HSP und Eintransferieren des Abbildes unterbleibt.

Beim HSP-ORG ersetzt der Neustart den Wiederanlauf. Damit kann das HSP-ORG auch von seriellen Datenträgern (Magnetbandkassetten) urgelen werden.

**Teilfunktionen** Beim Neustart des Systems werden unabhängig von der Ursache zu dessen Auslösung folgende Teilfunktionen durchlaufen:

- Rücksetzen aller peripheren Einheiten
- Einstellen des Anfangszustands der ORG-Listen
- Basisparametrieren, Parametrieren der Geräte
- Neuordnen des ORG-Listenpools
- Leeren der ORG-Warteschlangen
- Beenden aller Programme
- Löschen aller Koordinierungszählerlisten-Einträge
- Austragen aller Dateien (auch Prozeßdateien) aus der Dateinamensliste
- Übergabe der Neustartkennung
- Start des Wiederanlaufprogramms oder des SBP/Mini-SBP

Eine Datenträgerprüfung findet nicht statt.

**Auslösung** Es gibt drei Ursachen für einen Neustart des Systems, nämlich

- manueller Eingriff
- Spannungsausfall/Spannungswiederkehr
- Listenüberlauf

**Kennbit-  
übergabe** Sinngemäß gilt auch bei Neustart die in Abschnitt 11.1 gemachte Aussage bezüglich der an den Anwender übergebenen Wiederanlaufinformation.

**Wiederanlauf-  
programm,  
Neustartteil** Ein Neustart ohne Peripherenspeicherzugriff setzt die Hauptspeicherpräsenz eines intakten Wiederanlaufprogramms mit spezifischem Neustartteil voraus. Es ist unmittelbar nach dem (Erst-)Anlauf des Systems als HRP zu laden.

Beim HSP-ORG ist das Mini-SBP ein HRP, ein Wiederanlaufprogramm ist nicht zwingend notwendig.



---

Generieren      Der Hauptspeicherresidente Neustartbaustein muß mit einem Wunsch beim ORG-Generieren abgesetzt werden. Er schließt die Funktion "Wiederaufsetzen" aus.

### 11.3 Wiederaufsetzen

Zielsetzung      Datenkonsistenz im Hauptspeicher und auf Peripheriespeichern mit Direktzugriff.

Anlaß            Bei Dialogsystemen existiert die Forderung, das System bei Spannungswiederkehr an der Programmstelle fortzusetzen, an der der Spannungsausfall auftrat, sofern es sich um ein Programm in Anwenderenebene handelt.

Ablauf            Um dies zu erreichen, werden bei Spannungswiederkehr laufende Transfers mit seriellen Geräten (z.B. Drucker) mit entsprechenden Anzeigen abgebrochen, Transfers mit Geräten mit Direktzugriff (ORG-intern) wiederholt. Die ORG-Listen bleiben unverändert.

Auslösung        Die Funktion Wiederaufsetzen wird ausschließlich nach einer Netzunterbrechung am ZE-/Grund-Rahmen aktiviert, nicht aber, wenn lediglich die Spannung am Erweiterungs-Rahmen ausgefallen ist.

ORG-Meldung      Nach einem gelungenen Wiederaufsetzen wird eine ORG-Meldung ausgegeben. Innerhalb der Meldung wird die Netzunterbrechungsdauer ausgegeben.

ORG-/Geräte-Reaktionen      Auf das Ereignis ZE-Spannungsausfall und Wiederaufsetzen bei Spannungswiederkehr reagieren das ORG und die Standardperipherie unterschiedlich.

Unterbrochene PSD-Aufrufe werden zur Erhaltung der Datenkonsistenz vom ORG wiederholt. Bezüglich der Behandlung der übrigen Geräte s. 7 in /102/.

Generieren      Der Hauptspeicherresidente Baustein muß mit dem Wunsch "Wiederaufsetzen" beim ORG-Generieren abgesetzt werden. Er schließt die Funktion "Neustart" aus.

---

#### 11.4 Spannungsausfall am Erweiterungsrahmen

- Zielsetzung**      Verkehr mit Geräten der Standardperipherie (s. 5) bei Spannungseinbruch am Erweiterungsrahmen zur Vermeidung von Systemausfällen und Dateninkonsistenzen.
- Anlaß**              Bei größeren Anlagen mit umfangreicher Geräteausstattung reichen die Steckplätze des ZE-Rahmens (Zentraleinheitensbaugruppenträger) nicht aus. Durch Auslagern des Systembusses in einen E-Rahmen (Erweiterungsbaugruppenträger) ist es möglich, die Steckplatzanzahl zu erhöhen.
- E-Rahmen besitzen jedoch eine eigene - vom ZE-Rahmen unabhängige - Stromversorgung (s. /25/, /26/). Wenn sie ausfällt, darf es nicht zur Blockierung oder zum Ausfall des gesamten Systems kommen.
- Ausfallsicherung**      Das ORG stellt die Funktion "Ausfallsicherung" zur Verfügung, die E/A-Aufrufe an Geräte, die sich an einem E-Rahmen befinden, bei Spannungsausfall bearbeitet. Sie wird aktiviert bei E-Rahmen-Spannungsausfall für die ausgefallenen Rahmen; implizit wird sie aktiviert bei Spannungsausfall am ZE-Rahmen für alle E-Rahmen.
- Generieren, 6.1 in /103/**      Die Ausfallsicherung wird generiert, wenn Geräte an E-Rahmen angeschlossen sind; d.h. im /G-Parametersatz E/A-Adressen größer/gleich 'H=9xxx' vorkommen. Die Ausfallsicherung wird nicht generiert, wenn keine solchen E/A-Adressen vorkommen oder wenn sie der Anwender explizit mit dem Generierwunsch /W:NOPNAU abwählt (Gründe zum Abwählen: a) bei SICOMP M20 können Geräte mit E/A-Adressen  $\geq$  'H=9xxx' vorkommen, obwohl keine E-Rahmen installiert sind, b) in besonderen Fällen könnte die Ausfallsicherung auch bei vorhandenen E-Rahmen stören).
- Der Anwender kann die Ausfallsicherung gerätespezifisch mit einer Auftragspufferung oder ohne eine solche generieren. Dazu ist eine Pufferzeit im /G-Parametersatz zu nennen bzw. wegzulassen.

ORG-Reaktionen

- . bei Auftragspufferung Die Ausfallsicherung speichert bei Spannungsausfall die E/A-Aufrufe zwischen, um sie nach Spannungswiederkehr und Integration der Geräte in das System wieder fortzusetzen. Das Programm, das die E/A-Aufrufe abgegeben hat, merkt nichts vom Spannungsausfall außer der Zeitverzögerung.

Die Zeit, wie lange bei Ausfall gepuffert wird, ist begrenzt:

- einmal durch die gerätespezifische Generierangabe der Pufferzeit und
- zum ändern durch einen bestimmten Füllgrad des ORG-Pools.

Das ORG möchte verhindern, daß wegen vieler neu eintreffender, wartender E/A-Aufrufe der ORG-Pool überläuft und es dadurch zu einem Wiederanlauf kommt. Wenn der ORG-Pool überzulaufen droht, beendet deshalb die Ausfallsicherung das Puffern und zwar ebenso, als sei die Pufferzeit abgelaufen.

Muß das Puffern aus einem dieser beiden Gründe beendet werden, bevor die Spannung wiederkehrt, bricht das ORG die anstehenden E/A-Aufrufe wie folgt ab:

- Abbruch mit Anzeige 0,5 und Fehlermeldung ..NOAPS..ANZ=anz.. für einen aktiven Aufruf (dessen E/A-Tätigkeit durch den Spannungsausfall unterbrochen wurde).
- Abbruch mit Anzeige 0,1,5 und Fehlermeldung ..NOAPS..DET=4508.. für die tätigen Aufrufe (das sind die in der Transferwarteschlange wartenden E/A).
- Zurückweisung mit Anzeige 0,1,5 und Fehlermeldung ..NOAPS..DET=430E.. für die E/A-Aufrufe, die danach eintreffen.

Nach Spannungswiederkehr und Geräteintegration erfolgt wieder normale E/A-Bearbeitung.

- . ohne Auftragspufferung Ist keine Auftragspufferung generiert, behandelt das ORG betroffene E/A-Aufrufe so, als sei die Pufferzeit sofort nach dem Spannungsausfall abgelaufen. Die zum Ausfalls-Zeitpunkt aktiven oder tätigen oder danach eintreffenden E/A-Aufrufe werden wie oben beschrieben abgebrochen.

- . ohne Ausfallsicherung Wurde die Ausfallsicherung beim Generieren abgewählt, erhält das ORG von einem ausgefallenen Gerät keine Rückmeldung mehr, nachdem es einen E/A-Auftrag abgegeben hat. Für den Anwender bedeutet dies, daß ein E/A-Aufruf nicht/nie beendet wird; auch nicht nach Spannungswiederkehr. Dies gilt sowohl für aktive, für tätige als auch für danach eintreffende E/A-Aufrufe bezüglich des Ausfallzeitpunkts.

Der Aufruf "Gerät rücksetzen" (s. /105/) kann diese Aufrufe wieder zurücknehmen, damit sie den ORG-Pool nicht belasten.

- 
- Meldungen Neben den Aufruf-Fehlermeldungen meldet die Ausfallsicherung jeden Spannungsausfall am Standardmeldegerät. Nach Spannungswiederkehr erfolgt eine Wiederbereit-Meldung an den ausgefallenen Datensichtstationen und für andere ausgefallene Geräte (z.B. PSD) am Standardmeldegerät (s. 4.2.4 in /103/, dort auch weitere Meldungen).
- Geräte-  
Reaktionen Die Geräte reagieren bei Ausfall unterschiedlich. Zur Integration der Geräte nach Spannungswiederkehr parametriert das ORG automatisch alle ausgefallenen und erneut betriebsbereiten Geräte.

---

## 12 Ladefunktionen

Beim Laden wird ein Ladeobjekt im HSP oder im Wartebereich sowie ggf. in der Arbeitsform abgespeichert. Zugleich wird es in die Buchführung des ORG eingetragen und bleibt dort vermerkt, bis es gelöscht wird. Geladene Objekte sind zugriffs- bzw. startklar (ruhend). Das Laden kann in Phasen erfolgen, s.u. Funktionen.

**Lader** Es stehen 2 Lader als PRP in einem Alternativlaufbereich oder im Laufbereich 1 zur Verfügung:

- Grundsprachelader, Programmnummer 2, Programmname "MLADØ2", segmentiertes PRP, Lader mit vollem Funktionsumfang; nicht in einem HSP-ORG verfügbar.
- Arbeitsformlader, Programmnummer 3, Programmname "MLADØ3", lineares PRP, zum (schnellen Nach-)Laden von Programmen aus zuvor geladenen Arbeitsformen.

Aufrufe oder Kommandos stoßen den gewünschten Lader an. Die Priorität der Lader wird beim Generieren festgelegt. In einem PSD-ORG stehen beide, in einem HSP-ORG nur der Arbeitsformlader mit eingeschränkter Funktion zur Verfügung (s. 12.3). Die beiden Lader ermöglichen folgende optimale Ladestrategie: In einer Installationsphase werden weitgehend die Programme in Arbeitsformen geladen sowie die CD und CC mit dem erforderlichen Ladebinden/Koppeln. Im laufenden Betrieb genügt meist der Arbeitsformlader zum schnellen Nachladen und auch Löschen (s. 12.3).

**Ladeobjekte** Ladeobjekte sind die Programme (HRP, PRP), Common Codes (CC) und Common Data (CD). Sie können in Pakete (Paket-HRP, -PRP, -CD) oder selbständig (S-HRP, S-PRP, SHP, CC) oder in den Commonbereich (Bereichs-CD) geladen werden; Programme können linear oder segmentiert sein (s. 2.1). Das zu ladende Objekt muß in Grundsprache oder bereits als Arbeitsform vorliegen; auf beliebigem PSD, in einer Grundsprachebibliothek bzw. in einer Arbeitsformdatei.

**Anstoß** Zum Laden und Löschen stehen zur Verfügung:

- Aufrufe (Parameterblock und Laderblock), s. /101/
- Kommandos (mit Voreinstellungen), s. /103/

Eine Laderwarteschlange und Koordinierungszähler koordinieren die Laderanstöße. Bearbeitet der Grundsprachelader einen Auftrag in der Phase "Erstellen der Arbeitsform", so kann in dieser Zeit der Arbeitsformlader parallel einen zweiten Auftrag bearbeiten. Sonst kann nur ein Auftrag bearbeitet werden (jeder Lader nur einen Auftrag und nur ein Lader kann tätig sein, wenn der Grundsprachelader nicht gerade eine Arbeitsform erstellt). Bei Parallelarbeit können spätere Aufträge an den schnellen Arbeitsformlader einen Grundspracheladevorgang überholen.

Funktionen Gezielt anstoßbare Funktionen sind (s. Bild 12/1):

- a) Laden von HRP (linear) aus Grundsprache in den HSP und ggf. ins Abbild
- b) Laden von HRP aus Grundsprache in den HSP und ggf. ins Abbild mit Erstellen der Arbeitsform
- c) Laden von PRP (linear) aus Grundsprache in den Wartebereich
- d) Laden von PRP aus Grundsprache in den Wartebereich mit Erstellen der Arbeitsform
- e) Laden von HRP aus einer Arbeitsform in den HSP
- f) Laden von PRP aus einer Arbeitsform in den Wartebereich
- g) Laden von CD und CC aus Grundsprache in den HSP
- h) Einrichten von leeren CD
- i) Löschen von PRP, HRP, CC und CD (Arbeitsformen werden dabei nicht gelöscht).

Soll nur die Arbeitsform erstellt werden (für späteres Nachladen nach e/f), so lädt der Anwender (nach b/d) und löscht danach wieder das Programm (i).

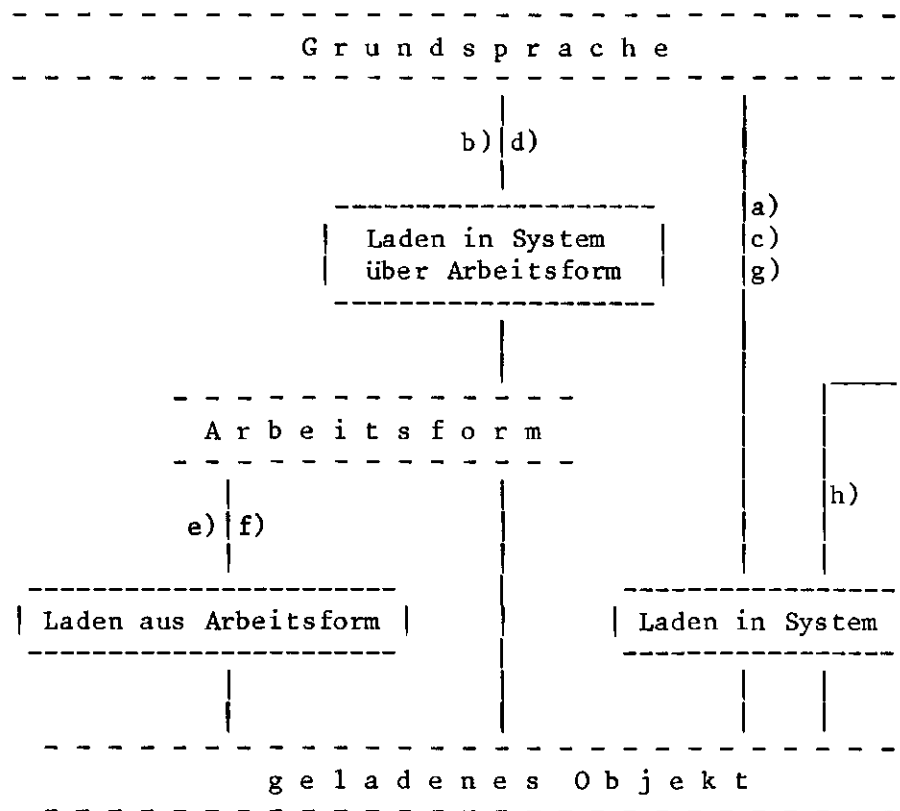


Bild 12/1 Ladefunktionen

---

## 12.1 Aufbau der Ladeobjekte

Ein Ladeobjekt kann sein:

- a) ein Grundsprachemodul ohne externe Adreßbezüge,
- b) ein segmentiertes Programm fertig gebunden,
- c) ein Objekt wie a) oder b), jedoch mit einseitig einbindbaren Adreßbezügen,
- d) ein bereits in einer Arbeitsform vorliegendes Programm.

|   |   |
|---|---|
| Grundsprache-<br>modul ohne<br>Externadressen | Er entstand entweder aus einem Assemblerabschnitt oder aus mehreren Abschnitten, die fertig gebunden wurden.  |
| segmentiertes<br>Programm                     | Es besteht aus mehreren Grundsprachemoduln einer Bibliothek: Nahtstellenphase und Segmente (s. auch 4.9).   |
| noch zu<br>bindendes<br>Objekt                | Ladeobjekte können noch Externadreßbezüge enthalten (Assembler-DEEX-Anweisungen für Externadreßdefinitionen und -NDEX-Anweisungen für aufgerufene Externadressen, X- und A-Sätze der Grundsprache), wenn diese sich durch einseitiges Binden (oder Ladebinden) befriedigen lassen (s. 12.3).                                  |
| Programm<br>bereits in<br>Arbeitsform         | Liegt ein Programm bereits in einer Arbeitsform vor, dann kann es aus dieser und deshalb schnell geladen werden. Der Ladeort kann dabei begrenzt neu gewählt werden; mehrfaches Laden ist möglich.  |
| Arbeitsform<br>. für segment.<br>Programme    | Beim Laden eines segmentierten Programms entsteht als Zwischenform die Arbeitsformdatei (Dateiname = Programmname). Sie enthält in Maschinensprache das bereits richtig adressierte, einseitig gebundene Programm; aufgeteilt in Buchhalter, Platz für ÜT und PT, Rootsegment, Segmente und Segmentwechsel-E/A-Teil (s. 4.9). |
| . für lineare<br>Programme                    | Auch lineare Programme können wahlweise in einer Arbeitsform zum Laden vorbereitet werden. Sie hat eine andere Struktur.  |

## 12.2 Ladeparameter

Ladeparameter sind Informationen aus dem Grundsprache-N-Satz des Programms und Parameter des Kommandos bzw. Aufrufs/Laderblocks.

|   |   |
|---|---|
| N-Satz-Parameter  | Abschnittsname (= Programmname), Programmzustandswort, Länge I-Teil, Länge V-Teil, Startadresse (bei fehlender Angabe wird am I-Teil-Anfang; fehlt dieser, am V-Teil-Anfang gestartet).   |
| Aufruf-/Kommando-Parameter<br>21.7:/101/<br>22.3:/101/<br>3.2:/103/ | <ul style="list-style-type: none"><li>- Ladertyp</li><li>- Identifikation des Ladeobjektes durch Gerät, Bibliothek und Element der Grundsprache</li><li>- Typ und Art des Objektes (PRP, HRP, CD, CC, Wiederanlaufprogramm und Serviceprogramm)</li><li>- Nummer und Priorität des Objekts</li><li>- Ladeort durch Laufbereich, Common Bereich, Paket und/oder Ladeadresse</li><li>- Längenreservierung, V-Teil-Verlängerung und Überlappungsprüfung</li><li>- Arbeitsformverwendung, -speicher und -namensmodifizierung</li><li>- Abbildbehandlung</li><li>- anzubindende und anzukoppelnde Objekte</li><li>- Bindemeldungen, Meldegerät</li></ul> |

Zur Handhabungsvereinfachung dienen bei bestimmten Parametern Voreinstellungs-, Vorbelegungs- und Rückmeldemechanismen (s. 22.3:/101/).

## 12.3 Lade/Lösch-Funktionen

|            |   |
|------------|---|
| Leistungen | Beim Laden und Löschen erbringt der Lader u.a. folgende Einzelleistungen: <ul style="list-style-type: none"><li>- Umsetzen von Grundsprache (GS) in Maschinensprache (MS)</li><li>- Ladebinden und Koppeln</li><li>- Erstellen der Arbeitsform</li><li>- Abspeichern des Objektes im HSP/Abbild oder Wartebereich und Startklarmachen</li><li>- Adressieren</li><li>- Platzvergabe</li><li>- Löschen</li><li>- Kontrollen, Anzeigen, Meldungen, Warnungen</li></ul> |
|------------|---|

Der Grundsprachelader erbringt alle Leistungen; Minderleistungen des Arbeitsformladers s.u.



- 
- Umsetzen  
GS in MS
- GS ist das Ergebnis des Übersetzens und Bindens; in einer Grundsprachebibliothek vorliegend. Maschinensprache ist der im HSP ablaufbare bzw. verarbeitbare Code. Der Lader erzeugt die MS aus der GS und legt die MS entweder im HSP (bei HRP, SHP, CD, CC) oder im Wartebereich (bei PRP) ab; zusätzlich kann eine Arbeitsform als MS-Zwischenform entstehen (s.u.). Der Lader erkennt am Grundspracheobjekt, ob es sich um ein segmentiertes oder um ein lineares Programm handelt. Liegen unter der gleichen Identifikation sowohl ein segmentiertes als auch ein lineares Programm vor, entscheidet der Anwender, welches er laden will.
- Ladebinden
- Ladebinden besteht aus dem einseitigen Binden der Adressen und dem Vereinigen der beiden ÜT (Einbinden der ÜT).
- Der Lader merkt sich die Externadreseffinitionen, die ein Modul enthält, in einem Externadreseffbuch (Jede Adresse, die von anderen Objekten benutzt wird, ist als symbolische Externadresse zu definieren). Enthält ein Modul Externadreseffaufrufe (als solche definierte symbolische Adressen), so erwartet der Lader diese im Externadreseffbuch. Er übernimmt die virtuellen Adressen aus dem gerufenen Objekt und ordnet sie den symbolischen Adressen im aufrufenden Objekt zu (dynamisches Binden). Mit mindestens einem Externadreseffaufruf wird das ganze Objekt eingebunden (alle Externadreseffdefinitionen werden zugeordnet).
- Einseitiges Binden setzt voraus, daß die fremden anzubindenden Objekte zuvor geladen wurden und die benötigten Adressen stets zugänglich sind. Das heißt im einzelnen:
- Externadreseffdefinitionen dürfen nur in Paket-HRP ohne Arbeitsform, in CD und CC vorkommen.
  - Externadreseffaufrufe dürfen in allen Ladeobjekten vorkommen; jedoch können Paketobjekte nur vom selben Paket gerufen werden.
- Zum Einbinden der ÜT übernimmt der Lader die Einträge in der ÜT des Commonbereichs (für gerufene CD) und trägt sie in das freie Ende der ÜT des Ladeobjekts (selbständige Programme). Reicht die "freie" ÜT nicht aus, um den Commonbereich ganz aufzunehmen, belegt der Lader nur den noch freien ÜT-Platz und gibt eine Warnung aus. Damit sind nur die am Ende des Commonbereichs geladenen CD ladegebunden. Die anderen CD bleiben fremder Adreseffraum.
- Die Paket-ÜT enthalten bereits Einträge der Commonbereichs-ÜT; damit liegen Paketprogramme, -CD und Bereichs-CD im selben Adreseffraum; letztere jedoch nur soweit die Paket-ÜT noch Platz hat. Es gilt das zuvor Beschriebene.

Eine besondere Form ist das explizite Einbinden eines Bereichs-CD oder des gesamten Commonbereichs beim Laden eines CC. Das explizite Einbinden wird durch Ladeparameter angegeben. Im Falle "CD einbinden" übernimmt der Lader nur die Externdefinitionen des angegebenen CD und die CB-ÜT-Einträge von diesem CD und den (im CB) folgenden CD. Im Falle "CB einbinden" übernimmt der Lader alle Externdefinitionen des CB und alle CB-ÜT-Einträge. Enthält der CC darüberhinaus auf andere Objekte gerichtete Externaufrufe, bleiben diese unbefriedigt; diese Objekte werden nicht eingebunden.

- Koppeln von CC Ruft ein Objekt einen CC, wird der CC wegen seiner besonderen Struktur nicht ladegebunden sondern gekoppelt (s. 2.4). Koppeln können S-HRP/PRP, CC selbst und Pakete. Das rufende Objekt übernimmt den CC-V-Teil durch Kopieren aus der CC-Datei und Anpassen der PT/ÜT und übernimmt die Einträge des CC-I-Teils in die Objekt-ÜT (Ladebinden der Adressen wie bei CD). Die Externadressen des Adreßbuchs (s. 12.4) übernimmt ein zu ladender CC (Erhöhung der Koppelstufe) bzw. verwendet ein zu ladendes Programm zum einseitigen Binden.
- . explizites Koppeln Ein selbständiges Objekt (HRP, PRP und CC) kann beim Laden explizit einen bestimmten CC ankoppeln; dieser ist Ladeparameter. Andere Externadressen werden nicht eingebunden/übernommen. Ein Programm als Ladeobjekt muß mindestens einen Externaufruf an den genannten CC enthalten.
  - . dynamisches Koppeln Ist beim Laden eines S-HRP/PRP kein CC explizit genannt, sucht der Lader den CC aus, der die Externaufrufe befriedigt (bei möglichst niedriger Koppelstufe) und koppelt ihn an.
  - . an Pakete koppeln Beim Laden von reentranten CC kann der Anwender Pakete angeben, in die noch kein Objekt geladen wurde. Der Lader entfernt dann die Seiteneinträge des CB aus der Paket-UT und trägt die des CC (I- und V-Teil) ein. Beim Laden der Paketobjekte werden die Externadressen zum CC gebunden; ausgenommen bei Arbeitsformobjekten. Werden Bereichs-CD zusätzlich von Paketobjekten benötigt, kann beim Laden des CC gleichzeitig der CB oder das Bereichs-CD explizit eingebunden werden (s.o.).
  - . Koppelstufen, CC-Schachtelung CC können zu anderen CC (explizit) koppeln, auch zu bereits gekoppelten. Ein CC, der zu einem ungekoppelten CC koppelt oder den Commonbereich bzw. einem Bereichs-CD explizit einbindet, erhält die Koppelstufe 1. Wird dieser CC wieder angekoppelt, erhält der koppelnde CC die Stufe 2. So sind bis 15 Stufen möglich. Dadurch kann sich der Anwender mehrere CC einschließlich CB zusammenstellen - genau die, die sein aufrufendes Programm benötigt - bevor er dieses lädt und dabei durch Auswahl eines CC alle benötigten CC koppelt. Die verschachtelten CC müssen untereinander keine Adreßbezüge haben.

| Bindemöglichkeiten | Bindeobjekt | A-<br>Art | selbständiges Ladeobjekt |     |     |    | Bindeform               |
|--------------------|-------------|-----------|--------------------------|-----|-----|----|-------------------------|
|                    |             |           | HRP                      | PRP | SHP | CC |                         |
| CD, m              | e           |           | X                        | X   | X   |    | dynamisches Ladebinden  |
| CD                 | 1           |           |                          |     |     | X  | explizites Ladebinden   |
| CB                 | 1           |           |                          |     |     | X  | explizites Ladebinden   |
| Paket              | 1           |           |                          |     |     | X  | expl. Koppeln an Pakete |
| CC                 | 1           | X         | X                        |     |     | X  | explizites Koppeln      |
| CC                 | e           | X         | X                        |     |     |    | dynamisches Koppeln     |

CD: Commonbereichs-CD

CC: mit beliebiger Koppelstufe

X: diese Form ist möglich

m: auch mehrere CD rufbar

1: Ladeparameter

e: Externadresaufwurf im Ladeobjekt

Bild 12/1 Bindeformen beim Laden von selbständigen Objekten

Erstellen der  
Arbeitsform

Segmentierte Programme müssen zwar vollständig in MS umgesetzt werden, können aber nicht vollständig in den HSP geladen werden. Deshalb erstellt der Lader eine Arbeitsform (s. 12.1), die alle umgesetzten Elemente aufnimmt, und fügt noch Elemente für Buchhaltungsinformationen und Segmentwechsel hinzu.

Mehrfaches (gleichzeitiges) Laden desselben Programms (unter verschiedenen Nummern) ist bei segmentierten Programmen möglich. Durch Aufruf/Kommando veranlaßt der Anwender die Modifizierung des Arbeitsformdateinamens (s. 22.3:/101/). Mit jedem Laden (mit neuer Programmnummer) wird dann eine eigene Arbeitsform erstellt und aus ihr weiter- und nachgeladen.

Auch für lineare Programme kann wahlweise (Ladeparameter) eine Arbeitsform erstellt werden.

Abspeichern  
und Startklar-  
machen

Damit ist das eigentliche Laden/Bereitstellen der MS in den HSP bzw. Wartebereich gemeint. Bei CD, CC und Programmen ohne Arbeitsform erfolgt dies direkt mit dem Umsetzen. Bei segmentierten und, wo gewünscht, bei linearen Programmen wird aus der zuvor erstellten Arbeitsform geladen. Existiert die Arbeitsform aus früheren Ladevorgängen, besteht ein Laden nur aus diesem Vorgang; diesen Sachverhalt ermittelt der Lader selbsttätig. Darüberhinaus kann der Anwender dieses schnelle Nachladen ausdrücklich wünschen (Ansprechen des Arbeitsformladers).

Zum Startklarmachen gehört der Aufbau von PT und UT und der Eintrag in die Objektbuchführung. Der Elementname des Ladeobjekts wird als Objektname zugeordnet.

Der Anwender kann veranlassen, daß das Objekt (HRP, CD) auch in das Abbild geladen wird.

Wiederholtes Nachladen aus derselben Arbeitsform an verschiedenen Stellen ist möglich. Das Programm ist zuvor zu löschen. Ein zuerst selbständig geladenes Programm kann nicht in ein Paket, ein Paketprogramm kann nur wieder in ein Paket nachgeladen werden (ein ladegebundenes sogar nur ins selbe Paket). Die Programme dürfen keine SAVE-Segmente besitzen (s. /213/).

**Adressierung** Der Lader überführt virtuelle Grundspracheadressen in virtuelle Maschinenadressen. Die mit der Grundspracheadresse 128 beginnenden V-Teile bleiben entweder unverändert (selbständige HRP, PRP und CC) oder werden auf die virtuelle Ladeadresse im Paket oder Commonbereich umgerechnet. Die mit der Grundspracheadresse 65535 endenden I-Teile werden ebenfalls so umadressiert, daß sie dem V-Teil folgen (nicht beim CC-I-Teil).

In der Arbeitsform ist die Adressierung bereits auf die Ladeadresse abgestimmt. Soll beim Laden aus der Arbeitsform auf eine andere Ladeadresse geladen werden, kann der Lader beim Abspeichern umadressieren (s.o.).

Zur Platzvergabe s. 12.4.

**Löschen** Die Löschfunktion bewirkt Austrag des Objektes aus den Listen der Programmorganisation und Freigabe des belegten Speicherplatzes in HSP und Wartebereich sowie Austrag aus dem Abbild, sofern dort vorhanden. Beim Löschen eines Programmes bleibt die Arbeitsform auf PSD erhalten. SHP werden dagegen vollständig gelöscht.

**Kontrollen** Der Lader kontrolliert Objekte und Parameter. Fehler führen zu Anzeigen und/oder Meldungen und meist zur Zurückweisung oder zum Abbruch des Ladens.

**Objekte laden** Folgende Einzelleistungen führt der Lader bei den verschiedenen Objekttypen durch:

- a) CD laden: Umsetzen und Adressieren, ggf. Ladebinden, Platzvergabe und Abspeichern
- b) CD einrichten: Buchführung und Platzvergabe
- c) CC laden: Umsetzen und Adressieren ggf. Ladebinden und Koppeln, Platzvergabe und Abspeichern
- d) Lineare Programme ohne Arbeitsform laden: Umsetzen und Adressieren, ggf. Ladebinden und (bei selbständigen) Koppeln, Platzvergabe, Abspeichern und Startklarmachen
- e) Programme mit Arbeitsform, Neuladen: Umsetzen und Adressieren, ggf. Ladebinden und (bei selbständigen) Koppeln, Erstellen der Arbeitsform, Platzvergabe, Abspeichern und Startklarmachen.
- f) Programm aus vorhandener Arbeitsform nachladen: Platzvergabe, Umadressierung, Abspeichern und Startklarmachen.
- g) SHP laden: Umsetzen und Adressieren, ggf. Ladebinden, Platzvergabe, Abspeichern und Startklarmachen.

Laden mit Arbeitsformlader Der Arbeitsformlader muß ausdrücklich aufgerufen werden, lädt schneller als der Grundsprachelader, hat jedoch einen geringeren Leistungsumfang:

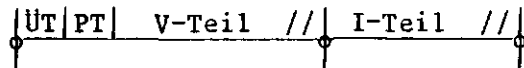
- Laden von HRP/PRP aus bestehenden Arbeitsformen (f)
- Einrichten von Paket-CD (b)
- Löschen von HRP, PRP und Paket-CD, kein Löschen von Externadreibezügen
- Kein Zugriff zum Abbild, kein Löschen von ins Abbild geladenen Objekten

In einem HSP-ORG, das nur den Arbeitsformlader hat, können nur selbständige HRP aus Arbeitsformen geladen und gelöscht werden. HRP dürfen nicht im Abbild vorliegen.

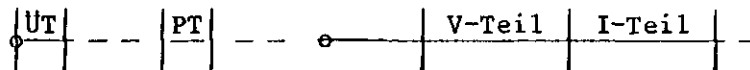
#### 12.4 Platzvergabe beim Laden

Die Platzvergabe für die einzelnen Ladeobjekte im HSP und auf PSD übernimmt teilweise das ORG, teilweise der Anwender. Die Ladeobjekte bilden folgende allgemeine Strukturen:

selbständige HRP Abzuspeichern vor dem ORG-Code-Bereich. Die Platzvergabe erfolgt durch den Lader.



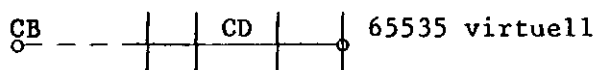
HRP in Paketen Die virtuelle Ladeadresse (Anfang des Programmumpfes, > 1023) ist vom Anwender zu nennen oder wird vom Lader ermittelt (Platzverwaltung). Es wird eine der PT der Seite 0 belegt und die Paket-ÜT mitbenutzt.



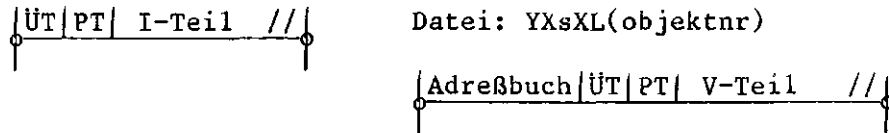
CD in Paketen Platzvergabe erfolgt durch den Anwender mit Nennung der virtuellen Anfangsadresse oder durch den Lader.



CD im Commonbereich Die Platzvergabe im Bereich übernimmt der Anwender durch Nennung der virtuellen Anfangsadresse oder der Lader. Die Anordnung der einzelnen CD im Bereich ist an der Adreßraumnutzung der rufenden Programme zu orientieren. Die ÜT steht im ORG.



**Common Code** Der CC wird beim Laden im HSP und auf PSD abgelegt. Im HSP steht nur der I-Teil mit ÜT/PT (auf Seiten gerastert), auf PSD stehen Adreßbuch, ÜT/PT und der V-Teil. Für jeden CC wird eine eigene Datei eingerichtet. Die Platzvergabe im HSP erfolgt durch den Lader analog den selbständigen HRP.



Das Adreßbuch enthält alle Externadreßdefinitionen, die beim Ankoppeln an ein Programm eingebunden werden. Ein CC höherer Koppelstufe beinhaltet die angekoppelten CC/CD/CB (Adreßbuch mit allen Definitionen, alle V-Teile, vereinigte Tafeln). Wird der CC im Modus REEN geladen, enthält der I-Teil im HSP den Code von V- und I-Teil; die Datei enthält nur Adreßbuch, ÜT und PT.

**PRP im Paket-Laufbereich** Der Anwender gibt die Laufbereichsnummer an. Jedes Paket-PRP benutzt die Paket-ÜT und den Platz einer PT in der Seite 0.



**selbständige PRP** Der Anwender gibt für den HSP keinen Speicherplatz, sondern die Laufbereichsnummer an bzw. gibt an, daß das ORG einen alternativen Laufbereich vorsehen soll. Selbständige PRP benutzen ihre eigene ÜT und PT. Struktur: wie selbständige HRP.

**segmentierte Programme** Anstelle des V/I-Teils steht bei segmentierten Programmen das Rootsegment mit der vom Lader beigegebenen Segmentwechselroutine und das Segmentfeld.

**SHP** Bei SHP (hauptspeichersegmentiertes HRP) enthält das Segmentfeld alle Segmente; die Wechselroutine entfällt.

**ÜT-Belegung, Speicherbelegung** Das Ladeobjekt belegt einen zusammenhängenden HSP-Platz. Die zugehörigen ÜT-Einträge erfolgen lückenlos; bei selbständigen Programmen am ÜT-Anfang, bei Paketobjekten entsprechend dem Paketplatz, bei Commonbereichs-CD entsprechend der Ladeadresse. Beim Ladebinden und Koppeln werden die ÜT-Einträge ergänzt (meist am ÜT-Ende). Die ÜT muß nicht lückenlos und aufsteigend belegt sein, auch können die über die ÜT ansprechbaren Speicherbereiche über dem ganzen HSP zerstreut liegen.

**Wartebereich** Den Platz im Wartebereich verwaltet allein der Lader.

---

### 13 Fehlerbehandlung

Treten Fehler im ORG-Ablauf, an Geräten oder in Anwenderprogrammen auf, die das ORG oder die Hardware erkennt, so werden entsprechend dem ermittelten Fehler Maßnahmen zur Weiterbehandlung des Anwenderprogramms und zur Fehlerbehandlung getroffen.

Unabhängig von der hier beschriebenen Fehlerbehandlung/-meldung geben ORG-Generator, Standardbedienprogramm und Wiederanlaufbaustein eigene Fehlermeldungen aus (s. /102, 103/).

**Fehlerursachen** Die erkannten und behandelten Fehler können folgende Ursachen haben:

- Gerätefehler (Parityfehler, Unklar, Übertragungsfehler u.a.). Diese können quittierbar sein, s.u.
- Programmierfehler (ORG erkennt fehlerhafte Aufrufe).
- Von der Hardware erkannte Fehler in den Programmen (nicht interpretierbare Befehle, Adressierfehler, Speicherfehler, Schreibverbot, Privilegverletzung, Adreßvergleich, arithmetische Fehler).
- Fehler im und beim Ablauf des ORG und der Grundprogramme.

**Fehlerreaktionen** Das ORG reagiert auf Fehler durch Anzeigen und Meldungen an den Benutzer, Bearbeiten von Quittungen und Maßnahmen zur Programmbehandlung.

. **Anzeigen** Der erkennende Funktionsbaustein teilt Fehler bei Aufrufen (Geräte- und Programmierfehler) dem Anwender als Anzeigen im Parameterblock mit (s. /101/).

. **Fehlermeldungen** Jeder Fehler führt zu einer Fehlermeldung, wenn dies nicht durch Sondermaßnahmen unterbunden wird. Die einzelnen Funktionsbausteine erkennen die Fehler und übergeben sie dem Fehlermeldebaustein zur Ausgabe an einem Meldegerät. Der Fehlermeldebaustein puffert Fehlermeldungen und gibt sie in der Reihenfolge ihres Auftretens aus. Aufbau und Komponenten der Meldungen s. Tabelle 13/1. Eine Liste der Fehlermeldungen enthält /103/.

Geräte-Fehlermeldungen können mit Quittungsmöglichkeit erfolgen. Die Zeichen "Q!" in der Fehlermeldung kennzeichnen, daß eine Quittung erwartet wird.

| Meldungskomponente       | Bedeutung  |
|--------------------------|--|
| @ datum uhrzeit          | Wenn dies (beim Generieren) gewünscht, meldet das ORG den Zeitpunkt des Erkennens des Fehlers in einer Vorab-Meldezeile.   |
| @                        | Die Meldung beginnt mit dieser Kennung (ORG-Meldung) und belegt eine Meldezeile.   |
| logger<br>datei logger   | Datenobjektidentifikation durch:<br>den logischen Gerätenamen bei Gerätefehlern oder den Dateinamen und den logischen Gerätenamen bei Fehlern im Zusammenhang mit Dateiverkehr.  |
| PR progrnr name<br>ORG   | Ablaufobjektidentifikation durch:<br>Programmnummer und -namen, wenn der Fehler in einem Programm erkannt wird oder<br>die Kennung ORG, wenn der Fehler im ORG auftrat bzw. keinem Programm zugeordnet werden konnte.  |
| PB-P adr<br>V            | Fehlerstelle durch:<br>die Adresse des Aufrufs (Parameterblocks), bei dem der Fehler auftrat oder  |
| BZ-P adr<br>V            | den Stand des Befehlsadreßregisters bei Auftreten des Fehlers.   |
| -P                       | Als Programmadresse, gekennzeichnet mit "-P", gibt das ORG die virtuelle Grundsprache-(Protokoll-)Adresse aus, wenn die Fehlerstelle im Rumpf (V-I-Teil) eines Programms liegt.  |
| -V                       | Die virtuelle Maschinenadresse, gekennzeichnet mit "-V", gibt das ORG aus, wenn die Fehlerstelle nicht in einem Programmrumpf liegt.   |
| BZ-V'adr                 | Befehlsadresse als virtuelle Maschinenadresse in einem fremden Adreßraum (nicht eingebundene Befehlsfolge).  |
| text                     | Globale Fehlerangabe als Kurztext oder Abkürzung, 6 Zeichen (z.B. UNKLAR, PROGRF).   |
| DET=detailnr<br>ANZ=panz | Genauere Fehleridentifizierung durch:<br>die Detailfehlernummer, 4 sedezimale Zeichen, bei Nicht-Gerätefehlern oder<br>die Peripherieanzeigen bei Gerätefehlern, die bei den E/A-Aufrufen übergeben werden.<br>panz: 4 sedezimale Zeichen = DEVPRB-Anzeigenwort<br>16 sedezimale Zeichen = PHYSPB-Anzeigenwörter |
| Q!;<br>AB;<br>;          | Abschluß der Meldung durch:<br>die Quittungsaufforderung, wenn Gerätefehler quittiert werden müssen, oder<br>die Programmabbruchkennung oder<br>die Kennung für Normalabschluß.  |

Die Kleinbuchstabenketten stehen für aktuelle Angaben

Tabelle 13/1: Meldungsaufbau und Komponenten



---

. Quittungen      Quittungen sind Antworten des Benutzers auf mit Quittungs-  
aufforderung "Q!" versehene Fehlermeldungen. Möglich sind

- Plus-Quittungen ( @ logger + ) oder
- Minus-Quittungen ( @ logger - ).

Quittungen ermöglichen, durch Eingriff des Systembedieners (z.B. Einschalten eines Gerätes) den Aufruf nachträglich noch auszuführen. Der Bediener entscheidet sich also entweder zur Behebung des Fehlerzustandes und zur Plus-Quittung und damit zum ordnungsgemäßen Weiterarbeiten des Aufrufs oder zur Minus-Quittung und damit zum Abbruch des Aufrufs mit Anzeigen. Siehe auch 5.3 und 9.2.

. Maßnahmen zur Programmbehandlung      Abhängig von der Fehlerursache kann das ORG neben der Übergabe von Anzeigen und Meldungen folgende Maßnahmen treffen:

- Abbruch des Aufrufs, Fortsetzung des Programms
- Warten auf Quittung, dann Programmfortsetzung mit Wiederholung oder Abbruch des Aufrufs
- Abbruch des Anwenderprogramms
- Ausführung der Wiederanlauf funktion
- Herstellung des STOP-Zustandes
- Verzweigung auf Anwender-Sonderoutine (s.u.)
- Meldungsbearbeitung durch die AMBOSS-Fehlerschnittstelle bzw. die Fehlermeldenachtstelle (s.u.)

Fehlermeldegeräte      Zur Ausgabe von Meldungen stehen dem Anwender das Fehlermeldegerät oder das Standardmeldegerät bzw. optionell das Ersatzgerät zur Verfügung. Zeitweise sind die für das Standardmeldegerät bestimmten Ausgaben auf ein anderes passendes Gerät umlenkbar (s.u. Bedien-/Meldegerät).

. Fehlermeldegerät      Als Fehlermeldegerät ist das Standardmeldegerät eingetragen; mit einem Kommando (/DEVCHANGE, /103/) kann der Anwender eine andere Datensichtstation oder einen Drucker als Fehlermeldegerät eintragen oder auch wieder auf das Standardmeldegerät zurückschalten. Die Einstellung gilt bis zum nächsten Wiederanlauf. Bei unklarem Fehlermeldegerät werden Fehlermeldungen auf dem Standardmeldegerät ausgegeben. Ist für ein Programm ein Bedien-/Meldegerät zugeordnet (Terminalbetrieb, /LOGON), wird dieses für die betreffenden Fehlermeldungen benutzt.

. Standardmeldegerät      Das Standardmeldegerät wird eingesetzt für jede Art von Systemmeldungen (s. 9) und gibt die Fehlermeldungen aus, wenn kein Fehlermeldegerät oder Bedien-/Meldegerät zugeordnet ist. Die Fehlermeldungen kommen auf dem Ersatzgerät, wenn das Standardmeldegerät unklar ist.

. Ersatzgerät      Kann eine Fehlermeldung nicht auf dem Standardmeldegerät ausgegeben werden, versucht das ORG die Ausgabe auf dem Ersatzgerät, falls generiert. Ist auch das Ersatzgerät unklar bzw. nicht generiert, unterbleibt die Meldung.

Gibt der Anwender Standard-E/A-Aufrufe an eines der Geräte ab, muß er beachten, daß Fehler minusquittiert und nicht gemeldet werden.

- 
- . Bedien-/ Meldegerät Ist ein Bedien-/Meldegerät zugeordnet (s. 9), übernimmt es alle Meldungen, die die ihm zugeordneten Programme verursachten, damit im Terminalbetrieb (s. 9.4) alle zugehörigen Meldungen am Terminal erscheinen.
  - . Fehlende Fehlermelde-  
möglichkeit Kann wegen defekten, nicht betriebsbereiten oder nicht vorhandenen Meldegeräts oder wegen bewußter Unterdrückung das ORG keine Fehlermeldung abgeben, so fährt es in gleicher Weise wie bei Ausgabe einer Meldung fort. Eine Quittung ist zwangsläufig nicht möglich, deshalb wird wie bei Minus-Quittung fortgesetzt.

Treten Fehlermeldungen (ohne Quittung) zu häufig auf, so daß der begrenzte Puffer überläuft, unterdrückt das ORG hinzukommende Meldungen so lange, bis alle gepufferten Meldungen ausgegeben sind. Danach teilt das ORG mit einer besonderen Meldung

@ anzahl LOST ERROR MESSAGES

ersatzweise die Anzahl der unterdrückten Meldungen mit.

Prüfungen unterdrücken, Programmzustand "ausgetestet"

Da die gründlichen Prüfungen in der Aufrufbearbeitung bei in Betrieb befindlichen und ausgetesteten Programmen in der Regel positiv ausgehen, besteht die Möglichkeit Plausibilitätsprüfungen zu unterdrücken, um die Aufrufbearbeitung zu beschleunigen. Dazu führt das ORG in der PNULI einem Zustand "ausgetestet". Der Anwender stellt diesen Zustand programmspezifisch ein und hebt ihn wieder auf durch Aufrufe (s. 21.9.4 in /101/) oder durch Kommandos (s. 3.1.19 und .20 in /103/); Aufrufe können ihn abfragen (s. 21.2.18 und .19 in /101/). Es liegt beim Anwender, ein Programm programmstückweise, zeitweise oder für immer als "ausgetestet" zu erklären. Dabei ist zu beachten, daß noch vorhandene Fehler zu ORG-Reaktionen führen: Aufheben des Zustandes "ausgetestet", Sammel-Anzeige/Meldung und Abbruch des Programms.

Anwender-  
nahtstellen  
zur Fehler-  
behandlung

Arithmetische Fehler in einem Programm kann der Anwender in einer Sonderroutine in diesem Programm abfangen und behandeln. Dazu adressiert der Anwender im Programm (PKEN-Anweisung) die Sonderroutine und Hilfszellen. Bei einem arithmetischen Fehler übergibt das ORG die Fehler-Befehlsadresse und das UAW-Bit (s. 40.1 in /101/) und verzweigt dann auf die Sonderroutine. Lag der Fehler in einem ausgelagerten Code (in einem fremden Adreßraum, Anwender-Adreßraumerweiterung /216/, so zeigt das ORG dies zusätzlich an und übergibt in den Hilfszellen den Tafelzeiger des fremden Adreßraums (TZ3).

Eine Fehlermeldenahtstelle zur Unterdrückung oder Erweiterung der Fehlermeldungen kann erstellt und ins ORG eingebracht werden (s. 3.4).

AMBOSS 4 (s. /500/) enthält immer die AMBOSS-Fehlerschnittstelle, die das Meldeverhalten an das Dialogsystem anpaßt.

|A|

Das Verzeichnis verweist auf Seitennummern

ablauffähiges Programm 36  
Ablaufobjekt 6, 23  
Ablaufzustand 36  
Adressierung, HSP 4.2  
- Laden 103  
Adreßraum 4, 4.1  
Adreßraumerweiterung 4.1, /216/  
Adreßräume, ORG 35  
aktuelles Programm 37  
Alarmbearbeitung, Funktion 64  
alternativer Laufbereich 45  
AMBOSS 4 0.1  
Analogeingabe 65  
Änderungsgenerierung 89  
angehaltenes Programm 36  
Anwender  
- Bedienprogramm 84  
- Nahtstelle 34  
- Schnittstelle, DUST 58  
- Schnittstelle, E/A 53  
- Simulation 73  
- Treiber 66  
Anzeigen 32, 107  
Arbeitsform 52, 99, 103  
Arbeitsformlader 97, 104  
Arithmetiksimulation 74  
Aufbau, Unterlage 0.2  
- ORG 1  
- Programm 24  
Aufruf 18, 31, /101/  
- Anzeige 31  
- Endebearbeitung 41  
- Funktion 18  
- Modifikation, E/A 51  
- Rückmeldungen 32  
- Struktur 18, 32  
- zeitmodifiziert 77  
Auftragspufferung 96  
Ausfallsicherung 96  
ausgetestet, Programmzustand 110  
Auslagerung, Blöcke 55  
Auslagerung, ORG-Teile 4.1  
Auslagerung, Subsysteme 4.1, /216/  
AWR = Adressierungsweichenregister 48

|B|

Baustein 13  
Bausteinablauffolge 4  
bedienbare Anwenderprogramme 84  
Bediengerät 80  
Bedienkommando 19  
Bedien-/Meldegerät 81, 107  
Bedienschutz 81

Bedienung 79  
Bedienungseinleitung 81  
Bedienungsübergabe 85  
Bedienzeichenersatz 86  
Beenden Programm 40  
Benutzer-Nahtstelle 9  
Betriebsmittelzuteilung 10  
Betriebssystem 0.1  
Bibliothek 28  
Binden 52, 101  
BS-M 0.1  
Buchführung 35

|C|

Cache 57  
CB = Commonbereich  
CC = Common Code  
CC-Schachtelung 102  
CD, Adreßraum 49  
CD = Common Data  
Cache 57  
Codeumwandlung 74  
Commonbereich 28  
Common Code 29  
Common Data 28  
Common-Data-Adresse ermitteln 47  
CPU-Zeitscheibe 44

|D|

Datei 7, 27  
- Anzahl 68  
- Art 27, 68  
- Buchführung, Dateiorganisation 68, 70  
- Ein-/Ausgabe, PSD 71  
- Identifikation 27, 68  
- Lage 68  
- Länge 68  
- Länge ändern 70  
- Organisation 67  
- organisatorische Funktionen 69  
- Schutz 69  
- Verkehr 10  
- Verwaltung 27  
- Verwaltungssystem 28  
- Zeiger 68  
Daten, gemeinsame 28  
- Art, E/A 54, 69  
- Austausch  
- Common Data 48  
- Feldübertragung 50  
- Programm 48

Datenfeld 55  
Daten, Pufferung 61  
- Satz, Dateiorganisation 68  
- Träger, Dateiorganisation 67  
- Wechsel, PSD 72  
- Vorbereitung, Dateiorganisation 67  
- Übergabe, Register 49  
- Übertragungssteuerung 58  
- Verwaltungssystem 28  
Datumeingabe 76  
Datumwechsel 77  
Disketten(-Laufwerke) 57  
DUST=Datenübertragungssteuerung 58  
DVS 28

E

E/A = Ein- und Ausgabe  
- DUST 58  
- Einheit 56.2  
- Magnetband 58  
- Magnetbandkassette 57  
- Organisation 53  
- Periphere Speicher 57  
- Verkehr 55  
- Verkehr, Koordinierung 54  
Einbringen, Objekt 9  
Einführung, ORG 1  
Einrichten, Datei 70  
einseitiges Binden 101  
Empfangen, Daten 50  
Ende Programm 40  
Erhöhen, Erniedrigen, Koordinierungszähler 43  
Eröffnen, Datei 70  
Ersatzgerät 109  
Erstellen der Arbeitsform 99  
explizites Warten 41  
Externadressen 99, 101

F

Fehler

- Behandlung 15, 107
- Meldegerät 109
- Meldung 107
- Nahtstellen 110
- Reaktion 56.2, 107
- Ursache 107

Feldeinheit 29

Felder, gemeinsame 28

Fernbedienung 80, 82

Festplatten 67

Firmwareverwaltung 25, /323/

fixieren Laufbereich 46

fortsetzen Programm 40

Funktion, ORG 9

G

GBS = Grundbetriebssystem 88

GEDA-Block 32, 55

Generieren 87, /102/, /501/

Generierparameter 89

Gerät, 9, 25

- Datenträger, Dateiorganisation 67
- globales 26

Gerätebaustein 56.2

Gerätegruppe 53

Geräteidentifikation 26

Gerätetyp 25

Geräteverwaltung 26

Gerätezuteilung 54

globale Geräte 26

Grenzprogrammnummer 81

Grundbetriebssystem 88

Grundprogramme 0.1

Grundsprache 98, 101

Grundsprachelader 97

Grundsprachemodul 99

H

Hardware-Ebenen 2

Hauptteil, ORG 1

Hilfsfunktion 16

Hilfsroutine 74

HRP 23

HRP, segmentiert 51

HSP, Speicherplatzaufteilung 4

HSP-ORG 89, 94

|I|

Identifikationskennung 22  
implizites Warten 41  
Inbetriebnahmeanleitung /102/  
integrierende Analogeingabe 65  
interne Funktion, ORG 4

|K|

Kennbitübergabe 94  
Kennungen 22  
Kommandos 18  
Kommandosprache 79  
Konvertierung 18, 74  
Koordinierung,  
- Funktion 39  
- Zähler 42  
Koppeln 102  
Kurzzeichenbedienung s. Bedienzeichenersatz 86  
Kurzzeitwecker 75

|L|

Ladebinden 101  
Ladefunktion 97  
Laden 52, 97  
Ladeobjekt 6, 97, 99  
Ladeparameter 100  
Lader 97  
Laufbereich 30  
- alternative 45  
Leistungsmerkmal, ORG 5  
Lesen, Feldübertragung 50  
Lieferform, ORG 5  
Lieferumfang, ORG 5, 88  
Listenpool 4  
L-Modus, Datenpufferung 62  
Logbuch 32  
lokale Netze 13  
Löschen 104  
Löschen, Datei 70

|M|

Magnetbandkassetten(-Laufwerke) 57  
Magnetbandlaufwerk 58  
Maschinenkennung 22  
Maschinensprache 101  
Mehrbenutzerbetrieb 85  
Mehrfach-Warten 41  
Meldung 32, 107  
Mini-SBP 83  
MK = Magnetbandkassette 57  
Monitornahtstelle 33  
Move s. Feldübertragung 50

|N|

Nahtstelle,

- Anwender 33
- Benutzer 9
- Monitor 33
- ORG 31

Netzwerk-Kommunikation 13

Neustart 94, /102/

|O|

Objekt 6

- Adresse ermitteln 47
- Parameter 32
- Zugriff 43, 44

Operateurverkehr 14

Operation 41

Operationsende, warten auf 41

ORG-M 0.1

ORG = Organisationsprogramm

- Anstoß 9
- Aufbau 1
- Bedienung 82
- Daten übergeben 47
- Dokumentation 0.1
- generieren 88
- internes Objekt 9
- Kennung 22
- nahe Programme 0.1
- Variante 88
- Zeit 76

|P|

Paket 29

Parameterübergabe 47

PB = Parameterblock 32, 55

periphere Speichereinheit 57

Peripherie 25

Peripherieanzeige 32, 108

Peripheralspeicher

- mit Direktzugriff 57
- Transfer 57

Platten(-Laufwerke) 57

Platzvergabe, Laden 105

Platzwechsel 37, 44

- sperren 44

Priorität 24

Prioritätsstruktur, ORG 1

Probegenerieren 88

Produktinformation 0.2



Programm 23

- Art 23
- beenden 39
- Ende 40
- fortsetzen 40
- Gerätezuordnung 85
- Nummernliste 35
- Organisation 35
- Parametertafel 35
- Parameterübergabe 47
- segmentiertes 51, 99
- starten 39
- Unterbrechbarkeit 25
- unterbrechen 40
- Wechsel anfordern 44
- Zustand 36, 110

Programmierhandbuch /101/  
programmierte Bedienung 85

Prozeß

- Peripherie 26, 53, 63
- Signalformer 64
- Zeitführung 77

PRP 23, 38

- fixieren Alternativ-Laufbereich 46
- segmentiert 41
- sperren Alternativ-Laufbereich 46
- Zeitscheibe 44

PSD-Cache 57, /107/

PSD = Peripheralspeicher mit Direktzugriff 57, 67

PT = Parametertafel 35

Pufferung 61

| $\bar{Q}$ |

Quittung 82, 109

| $\bar{R}$ |

Räumgewicht 37

Reaktionszeit 12

regenerieren, Daten 16

Relativzeit-Zähler 78

retten, Daten 16

Rootsegment 52

RPZ = Befehl 32

ruhend 36

Rückmeldung, ORG 32

Rücktransfer, PRP 38

S

SBP = Standardbedienprogramm 82  
SCC = Common Code = CC  
schließen, Datei 70  
Schreiben, Feldübertragung 50  
schützen, Objekt 15  
Schutzfunktion, Dateiorganisation 69  
segmentiertes Programm 10, 23, 51  
Segmentierung 4  
Senden, Daten 50  
Service 16  
Service-Unterstützung 56.1  
SHP, segmentiert 23, 48  
SICOMP M 0.1  
Sicherungsfunktion 15  
Simulation 73  
Simulation ZE 03 73  
Simultanmarbeit 11  
S-Modus, Datenpufferung 52  
Software-Betriebsmittel 8  
Software-Priorität 24  
Sonderperipherie 53, 66  
Spannungsausfall, E-Rahmen 96  
Speicher  
- Belegung 106  
- Platzaufteilung, HSP 3  
- Platzverwaltung/vergabe 8  
- Schutz 4.1  
- Tausch 93  
sperrern, Laufbereich 46  
Standard  
- Bedienung 82  
- Bedienprogramm 82  
- Ein-/Ausgabe 54  
- Geräteverkehr 56  
- Meldegerät 80, 109  
- Peripherie 26, 53  
starten 39  
Startsperre 39  
Steuerzeichen 56  
Struktur, Laufbereich 45  
Strukturparameter 5  
System  
- Aktivierung 91  
- Datei 89  
- Erstellung 87  
- Inbetriebnahme 16  
- Information 17  
- Kennzeichen 22, 88  
- Nahtstelle 17  
- Simulation 17, 73  
- Software 0.1  
- Speicherwechsel 93  
- Umgebung, ORG 5

|T̄|

Tätig-Bit-Abfrage 42  
Tafelzeiger 4.1  
Tafelzeiger kellern 74  
Teleservice 80  
Terminalbetrieb 85  
Testnahtstelle 33  
Testparameterübergabe 47  
Transfer 54, 71  
Transferwarteschlange 55  
Treiber 13, 57  
TZ(R) = Tafelzeiger(register) 4.1, 46

|Ū|

Überlagerungsstruktur, segmentiertes Programm 51  
Übersetzungstafel 35  
Übersicht, System 0.1  
Überwachen 15  
Uhr,  
- gepufferte 75  
- Software- 76  
Umsetzen, GS in MS 101  
Umstrukturierung, alternativer Laufbereich 45  
Unterbrechbarkeit 25  
Unterbrechung  
- Ebene 12  
- Ereignisse 12  
- Programm 40  
- System 12  
ÜT = Übersetzungstafel 35  
ÜT-Belegung 106

|V̄|

Verbindlichkeit der Unterlage 0.2  
Verkehrsablauf, E/A 13  
Verständigungsbereich 4  
Verwaltungsobjekt 6, 23  
Verweilzeit 44  
virtuelle  
- Adressierung 4.2  
- Gerät 26  
Vorwort 0.2

W

Wartebereich 7, 30  
Warten 41  
wartendes Programm 36  
Warteschlangen 35  
Wartungshilfe EVITAM 90  
Wechselplatte 67  
Wecker 75  
Wiederanlauf 92, /102/  
Wiederaufsetzen 95  
WS = Warteschlange

Z

Zähler, Koordinierungs 42  
Zeichenvorrat 56  
Zeit  
- Aktualisierung 76  
- Funktion 17, 75  
- Optimierung, Dateiorganisation 71  
- Scheibenbearbeitung 44  
- Überwachung, Geräte 56.1  
- Warteschlange 78  
Zeitgeber 75  
zeitmodifizierte Aufrufe 77  
Zielsetzung, Unterlage 0.2  
Zugriff, Objekte 48  
Zugriffsarten, Datei 69  
Zusatzfunktion 16  
Zustandsmodifikation 37  
Zustandswechsel 36  
Zuteilen 54, 82

Ausgabe, Änderung, Datum

| Seite      | Ausgabe 02<br>7.85<br>AE | Ausgabe 03<br>5.87<br>AE | Ausgabe 04<br>8.89<br>AE |
|------------|--------------------------|--------------------------|--------------------------|
| Titelblatt | 7.85                     | 5.87                     | 8.89                     |
| 0.1 - 0.2  |                          |                          | 3                        |
| 0.3        | 1                        |                          |                          |
| 0.4        |                          |                          | 2                        |
| 0.5        | 1                        |                          |                          |
| 0.6 - 0.9  |                          |                          | 3                        |
| 0.10       |                          |                          | 1                        |
| 1          | 1                        |                          |                          |
| 2          |                          |                          | 2                        |
| 3 - 4      |                          |                          | 3                        |
| 4.1 - 4.2  |                          |                          | 2                        |
| 4.3        |                          |                          | 0                        |
| 5          |                          |                          | 3                        |
| 6 - 8      |                          | 2                        |                          |
| 9          |                          | 1                        |                          |
| 10 - 12    | 0                        |                          |                          |
| 13 - 15    |                          | 2                        |                          |
| 16 - 17    | 1                        |                          |                          |
| 18 - 22    |                          | 2                        |                          |
| 23         | 0                        |                          |                          |
| 24 - 25    |                          | 2                        |                          |
| 26         |                          |                          | 2                        |
| 27         | 0                        |                          |                          |
| 28         | 1                        |                          |                          |
| 29         |                          | 2                        |                          |
| 30         |                          | 1                        |                          |
| 31         | 0                        |                          |                          |
| 32         | 1                        |                          |                          |
| 33         |                          |                          | 3                        |
| 34         | 1                        |                          |                          |
| 35         |                          |                          | 3                        |
| 36         | 1                        |                          |                          |
| 37         | 0                        |                          |                          |
| 38 - 39    | 1                        |                          |                          |
| 40         | 0                        |                          |                          |
| 41         | 1                        |                          |                          |
| 42         | 0                        |                          |                          |
| 43         | 1                        |                          |                          |
| 44         | 0                        |                          |                          |
| 45 - 46    |                          | 1                        |                          |
| 47 - 49    |                          | 2                        |                          |
| 50         |                          |                          | 2                        |
| 51 - 52    | 0                        |                          |                          |
| 53 - 54    | 1                        |                          |                          |
| 55         |                          | 2                        |                          |
| 56         | 1                        |                          |                          |
| 56.1 - .2  |                          |                          | 1                        |

Ausgabe, Änderung, Datum

| Seite     | Ausgabe 02<br>7.85<br>AE | Ausgabe 03<br>5.87<br>AE | Ausgabe 04<br>8.89<br>AE |
|-----------|--------------------------|--------------------------|--------------------------|
| 57 - 58   |                          |                          | 3                        |
| 59 - 60   | 0                        |                          |                          |
| 61        |                          |                          | 1                        |
| 62 - 65   | 0                        |                          |                          |
| 66        | 1                        |                          |                          |
| 67 - 69   | 0                        |                          |                          |
| 70        | 1                        |                          |                          |
| 71 - 72   | 0                        |                          |                          |
| 73        |                          |                          | 1                        |
| 74        |                          |                          | 2                        |
| 75        | 0                        |                          |                          |
| 76 - 77   | 1                        |                          |                          |
| 78        | 0                        |                          |                          |
| 79        | 1                        |                          |                          |
| 80 - 82   |                          | 2                        |                          |
| 83 - 85   | 1                        |                          |                          |
| 86        |                          | 2                        |                          |
| 87 - 89   |                          |                          | 2                        |
| 89.1      |                          |                          | 0                        |
| 90        |                          |                          | 2                        |
| 90.1      |                          |                          | 0                        |
| 91 - 92   | 1                        |                          |                          |
| 93        |                          |                          | 3                        |
| 93.1      |                          |                          | 0                        |
| 94        | 1                        |                          |                          |
| 95        |                          | 2                        |                          |
| 96        |                          | 1                        |                          |
| 96.1 - .2 |                          | 0                        |                          |
| 97 - 100  | 1                        |                          |                          |
| 101 - 102 |                          | 2                        |                          |
| 103       |                          |                          | 3                        |
| 104 - 106 |                          | 2                        |                          |
| 107       | 1                        |                          |                          |
| 108       |                          | 2                        |                          |
| 109       | 0                        |                          |                          |
| 110       |                          | 1                        |                          |
| 111 - 112 |                          |                          | 3                        |
| 113 - 116 |                          | 2                        |                          |
| 117       |                          |                          | 3                        |
| 118       |                          | 2                        |                          |
| 119 - 120 |                          |                          | 3                        |
| 121       |                          |                          | 3                        |
| 122       |                          |                          | 1                        |